

# Máster Interuniversitario en Estadística e Investigación Operativa UPC-UB

**Título:** Estimar mediante simulación el inventario a comprar considerando predicciones de la demanda de los productos

**Autor:** Ana Mogrovejo Del Salto

**Director:** Dr. Pau Fonseca i Casas

**Departamento:** Departamento de Estadística e Investigación Operativa

**Universidad:** Universidad Politécnica de Catalunya

**Convocatoria:** 2017-2018



## Índice general

<b>ÍNDICE GENERAL .....</b>	<b>2</b>
<b>ÍNDICE DE ILUSTRACIONES.....</b>	<b>4</b>
<b>INTRODUCCIÓN .....</b>	<b>6</b>
<b>OBJETIVO .....</b>	<b>6</b>
<b>RESEÑA DE LA EMPRESA COFEGO S.A. ....</b>	<b>6</b>
<b>HIPÓTESIS DEL MODELO .....</b>	<b>7</b>
HIPÓTESIS SIMPLIFICADORAS .....	7
HIPÓTESIS SISTÉMICAS DE DATOS.....	8
<b>DATOS .....</b>	<b>9</b>
SELECCIÓN DE PRODUCTOS A SIMULAR .....	9
DISTRIBUCIÓN DE PROBABILIDAD DE LOS DATOS .....	10
<b>SISTEMA REAL.....</b>	<b>23</b>
<b>SDL.....</b>	<b>24</b>
<b>MODELO EN SDL SISTEMA REAL.....</b>	<b>25</b>
<b>SISTEMA COFEGO .....</b>	<b>25</b>
BLOQUE COLA SERVIDOR VENDEDOR .....	26
BLOQUE SERVIDOR VENDEDOR .....	26
BLOQUE COLA SERVIDOR BODEGA .....	26
BLOQUE SERVIDOR BODEGUERO .....	27
PROCESO COLA SERVIDOR VENDEDOR.....	28
PROCESO SERVIDOR VENDEDOR.....	31
PROCESO COLA SERVIDOR BODEGA .....	38
PROCESO SERVIDOR BODEGUERO.....	42
<b>ARENA .....</b>	<b>46</b>
<b>MODELO EN ARENA SISTEMA ACTUAL.....</b>	<b>47</b>
<b>MODELO DE SISTEMA DE CONTROL INVENTARIO PERIÓDICO .....</b>	<b>53</b>
<b>MODELO EN ARENA SISTEMA MEJORADO .....</b>	<b>54</b>
<b>DISEÑO DE EXPERIMENTOS .....</b>	<b>56</b>
<b>ANOVA 2 FACTORES FIJOS .....</b>	<b>58</b>
ANÁLISIS DE LA TABLA ANOVA .....	58

VALIDACIÓN DE HIPÓTESIS DEL ANOVA .....	59
<b>VERIFICACIÓN Y VALIDACIÓN .....</b>	<b>60</b>
<b>RECOMENDACIONES Y CONCLUSIONES .....</b>	<b>61</b>
<b>BIBLIOGRAFÍA .....</b>	<b>63</b>

## Índice de Ilustraciones

ILUSTRACIÓN 1, “SELECCIÓN DE PRODUCTOS A SIMULAR” .....	10
ILUSTRACIÓN 2, “DIAGRAMA DE LA DISTRIBUCIÓN DEL PRODUCTO 1 (I1174)” .....	11
ILUSTRACIÓN 3, “CUADRO RESUMEN DE ESTADÍSTICOS DESCRIPTIVOS DEL PRODUCTO 1 (I1174)” .....	12
ILUSTRACIÓN 4, “DIAGRAMA DE LA DISTRIBUCIÓN DEL PRODUCTO 2 (I2793)” .....	12
ILUSTRACIÓN 5, “CUADRO RESUMEN DE ESTADÍSTICOS DESCRIPTIVOS DEL PRODUCTO 2 (I2793)” .....	13
ILUSTRACIÓN 6, “DIAGRAMA DE LA DISTRIBUCIÓN DEL PRODUCTO 3 (I1175)” .....	13
ILUSTRACIÓN 7, “CUADRO RESUMEN DE ESTADÍSTICOS DESCRIPTIVOS DEL PRODUCTO 3 (I1175)” .....	14
ILUSTRACIÓN 8, “DIAGRAMA DE LA DISTRIBUCIÓN DEL PRODUCTO 4 (I2245)” .....	14
ILUSTRACIÓN 9, “CUADRO RESUMEN DE ESTADÍSTICOS DESCRIPTIVOS DEL PRODUCTO 4 (I2245)” .....	15
ILUSTRACIÓN 10, “DIAGRAMA DE LA DISTRIBUCIÓN DEL PRODUCTO 5 (I3567)” .....	15
ILUSTRACIÓN 11, “CUADRO RESUMEN DE ESTADÍSTICOS DESCRIPTIVOS DEL PRODUCTO 5 (I3567)” .....	16
ILUSTRACIÓN 12, “DIAGRAMA DE LA DISTRIBUCIÓN DEL PRODUCTO 6 (I1429)” .....	16
ILUSTRACIÓN 13, “CUADRO RESUMEN DE ESTADÍSTICOS DESCRIPTIVOS DEL PRODUCTO 6 (I1429)” .....	17
ILUSTRACIÓN 14, “DIAGRAMA DE LA DISTRIBUCIÓN DEL PRODUCTO 1 (I1174)” .....	18
ILUSTRACIÓN 15, “CUADRO RESUMEN DE ESTADÍSTICOS DESCRIPTIVOS DEL PRODUCTO 1 (I1174)” .....	18
ILUSTRACIÓN 16, “DIAGRAMA DE LA DISTRIBUCIÓN DEL PRODUCTO 2 (I2793)” .....	19
ILUSTRACIÓN 17, “CUADRO RESUMEN DE ESTADÍSTICOS DESCRIPTIVOS DEL PRODUCTO 2 (I2793)” .....	19
ILUSTRACIÓN 18, “DIAGRAMA DE LA DISTRIBUCIÓN DEL PRODUCTO 3 (I1175)” .....	20
ILUSTRACIÓN 19, “CUADRO RESUMEN DE ESTADÍSTICOS DESCRIPTIVOS DEL PRODUCTO 3 (I1175)” .....	20
ILUSTRACIÓN 20, “DIAGRAMA DE LA DISTRIBUCIÓN DEL PRODUCTO 4 (I2245)” .....	21
ILUSTRACIÓN 21, “CUADRO RESUMEN DE ESTADÍSTICOS DESCRIPTIVOS DEL PRODUCTO 4 (I2245)” .....	21
ILUSTRACIÓN 22, “DIAGRAMA DE LA DISTRIBUCIÓN DEL PRODUCTO 5 (I3567)” .....	22
ILUSTRACIÓN 23, “CUADRO RESUMEN DE ESTADÍSTICOS DESCRIPTIVOS DEL PRODUCTO 5 (I3567)” .....	22
ILUSTRACIÓN 24, “DIAGRAMA DE LA DISTRIBUCIÓN DEL PRODUCTO 6 (I1429)” .....	23
ILUSTRACIÓN 25, “CUADRO RESUMEN DE ESTADÍSTICOS DESCRIPTIVOS DEL PRODUCTO 6 (I1429)” .....	23
ILUSTRACIÓN 26, “SISTEMA COFEGO” .....	25
ILUSTRACIÓN 27, “BLOQUE COLA SERVIDOR VENDEDOR” .....	26
ILUSTRACIÓN 28, “BLOQUE SERVIDOR VENDEDOR” .....	26
ILUSTRACIÓN 29, “BLOQUE COLA SERVIDOR BODEGA” .....	26
ILUSTRACIÓN 30, “BLOQUE COLA SERVIDOR BODEGUERO” .....	27
ILUSTRACIÓN 31, “PROCESO COLA SERVIDOR VENDEDOR” .....	28
ILUSTRACIÓN 32, “PROCESO SERVIDOR VENDEDOR” .....	35
ILUSTRACIÓN 33, “PROCESO COLA SERVIDOR BODEGA” .....	38
ILUSTRACIÓN 34, “PROCESO SERVIDOR BODEGUERO” .....	44
ILUSTRACIÓN 35, “MODELO EN ARENA SISTEMA ACTUAL” .....	48
ILUSTRACIÓN 36, “MODELO EN ARENA SISTEMA MEJORADO” .....	55
ILUSTRACIÓN 37, “DISEÑO DE EXPERIMENTOS” .....	57
ILUSTRACIÓN 38, “REPLICAS DEL DISEÑO DE EXPERIMENTOS” .....	57
ILUSTRACIÓN 39, “INTERVALOS DE CONFIANZA” .....	57

ILUSTRACIÓN 40, “TABLA ANOVA” .....	58
ILUSTRACIÓN 41, “TEST DE TUKEY” .....	58
ILUSTRACIÓN 42, “GRÁFICO RESIDUOS VERSUS VALORES PREDICHOS” .....	59
ILUSTRACIÓN 43, “QQ PLOT” .....	60

## **Introducción**

Para que una empresa asegure su competitividad en el mercado y la satisfacción de sus clientes, uno de los factores críticos que le ayudará a lograr sus objetivos es el correcto manejo de su inventario. Actualmente es indispensable que una empresa cuente con políticas adecuadas para el manejo del inventario, de lo contrario se pueden generar diversas consecuencias negativas para la empresa tales como ventas perdidas, grandes costos de orden y mantenimiento de inventario, pérdida de aceptación por parte de los clientes, entre otras. Uno de los objetivos de las políticas de inventario es determinar cuándo comprar y qué cantidad pedir a fin de evitar roturas de stock o un sobre stock de productos.

COFEGO S.A. ha experimentado problemas con el manejo de su inventario, el cual a lo largo de la trayectoria de la empresa se ha llevado a cabo de forma empírica, por lo que el presente trabajo está orientado a implementar políticas de inventario. Se propone un sistema de control de inventario periódico, debido a que COFEGO al ser una empresa pequeña sus proveedores son pocos por lo cual se ha considerado que es mejor realizar pedidos periódicos que agrupen todos los productos que la empresa requiere hasta el siguiente periodo de revisión. Para determinar la cantidad óptima a comprar y la amplitud del periodo se realiza un modelo de simulación en Arena, el cual ayudará a dar respuesta a la problemática simulando los diversos escenarios y permitiendo escoger el que mejor se adapte a las necesidades de la empresa.

## **Objetivo**

El objetivo del presente proyecto es estimar mediante una simulación la cantidad de productos a comprar y la amplitud del periodo en que se debe comprar basándose en la demanda de los productos, con la finalidad de que no se tenga un sobre stock de productos que generen una inmovilización de recursos tales como espacio en bodega, capital, etc. o una falta de stock que ocasione ventas perdidas.

## **Reseña de la empresa COFEGO S.A.**

COFEFO S.A. es una empresa de tipo familiar, ubicada en Machala, Ecuador; se dedica a la compra, venta y distribución de productos de consumo masivo y escolar, actualmente cuentan con 15 colaboradores que laboran en las distintas áreas de la empresa tales como: Bodega, Ventas, Administración, Contabilidad, entre otras.

Actualmente es su propietario el Ing. Fernando González, quien desempeña el cargo de Gerente General y se encarga de realizar los pedidos a los proveedores, a diario revisa el inventario de cada producto y basado en su conocimiento del negocio y experiencia determina la cantidad de cada producto a comprar a los proveedores, esto lo realiza de forma empírica sin seguir ninguna política. Es importante también destacar que la empresa no dispone de indicadores que midan su nivel de servicio, sin embargo, han notado que sus ventas perdidas a causa de falta de inventario son cada vez más frecuentes.

## **Hipótesis del modelo**

### **Hipótesis simplificadoras**

Hps1: La empresa ofrece solamente 7 productos.

Estado de validación de la hipótesis: Validada, ya que el objetivo del proyecto es determinar la cantidad de productos a comprar necesaria para atender la demanda el hecho de seleccionar solo 7 productos no tendrá inferencia en los resultados de la simulación, los resultados se enfocan en realizar recomendaciones en base solo a los 7 productos críticos que han sido previamente seleccionados.

Hps2: El vendedor y bodeguero trabajan 8 horas seguidas.

Estado de validación de la hipótesis: Validada, ya que dicha suposición no tendrá inferencia en los resultados de la simulación, a pesar de que en la realidad trabajan 8 horas laborables, pero tienen 1 hora de almuerzo.

Hps3: El bodeguero solo se dedica a despachar los pedidos de los clientes y a receptar las entregas de los proveedores de los 7 productos seleccionados.

Estado de validación de la hipótesis: Validada, ya que esta suposición no tendrá inferencia en los resultados de la simulación puesto que no es del interés del estudio analizar la ocupación del bodeguero, aunque en realidad el bodeguero realice adicionalmente otras actividades tales como ordenar la bodega, conteo del inventario de productos, recibir suministros de oficina, recibir y despachar otros productos que no han sido seleccionados para la simulación, etc.

Hps4: El vendedor solo se dedica a vender los 7 productos seleccionados.

Estado de validación de la hipótesis: Validada, ya que esta hipótesis no tendrá inferencia en los resultados de la simulación puesto que no es del interés del estudio analizar la ocupación del vendedor, aunque en realidad el vendedor realice adicionalmente otras actividades tales como venta de otros productos que no han sido seleccionados para la simulación, realización y envío de reportes diversos, atención a proveedores para compra de nuevos productos, etc.

### **Hipótesis sistémicas de datos**

Hp1: El tiempo de arribo entre clientes sigue una distribución exponencial con media 47.80 en unidades de minutos.

Estado de validación de la hipótesis: No validada, sin embargo, al observar los resultados de la simulación se observa que la distribución supuesta es coherente con la realidad del sistema.

Hp2: El tiempo de arribo entre las entregas de los proveedores sigue una distribución exponencial con media 2.50 en unidades de día.

Estado de validación de la hipótesis: No validada, sin embargo, al observar los resultados de la simulación se observa que la distribución supuesta es coherente con la realidad del sistema.

Hp3: El tiempo de servicio del vendedor sigue una distribución exponencial con media 10 en unidades de minutos.

Estado de validación de la hipótesis: No validada, la distribución fue determinada basándose en datos provistos por el vendedor, sin embargo, al observar los resultados de la simulación se observa que la distribución supuesta es coherente con la realidad del sistema.

Hp4: El 98% de los clientes cuando no hay inventario suficiente para atender su pedido decide comprar el inventario actual de los productos.

Estado de validación de la hipótesis: No validada, el porcentaje fue determinado en base al conocimiento y experiencia del vendedor, a pesar de esto se observa que los resultados de la simulación son coherentes con la realidad.

Hp5: El tiempo de servicio del bodeguero sigue una distribución exponencial con media 12 en unidades de minutos.

Estado de validación de la hipótesis: No validada, la distribución fue determinada basándose en datos provistos por el bodeguero, sin embargo, al observar los resultados de la simulación se observa que la distribución supuesta es coherente con la realidad del sistema.



## **Datos**

Los datos son cruciales para que la validación del modelo y el diseño de experimentos den buenos resultados, es por esta razón que su análisis es muy importante. Los datos utilizados fueron entregados por la empresa, son reportes procedentes del sistema “Otisa”, adicionalmente la empresa cuenta con manuales de procesos y funciones en los que se detalla como cada persona debe ingresar datos al sistema, por lo que pueden considerarse datos confiables. Los datos suministrados corresponden al año 2017 completo y se entregaron los siguientes archivos:

- Se ha proporcionado un maestro de inventario, que detalla: el código del producto, la descripción del producto, si el producto se vende por unidad o caja, cuantas unidades tiene una caja, el costo unitario del producto, el precio de venta del producto, etc.
- Detalle de las facturas de ventas por producto, que detalla: fecha, número de factura, código de cliente, nombre de cliente, apellido del cliente, el código del producto, la descripción del producto, cantidad vendida del producto, subtotal (precio unitario del producto por cantidad), descuento, IVA (Impuesto al Valor Agregado), total (subtotal – descuento + IVA), etc.
- Detalle de las facturas de compras por producto, que detalla: Fecha, número de factura, código de proveedor, razón social del proveedor, código del producto, descripción del producto, cantidad comprada del producto, subtotal (costo unitario del producto por cantidad), descuento, IVA, total (subtotal – descuento + IVA), etc.

## **Selección de productos a simular**

COFEGO S.A. ofrece a sus clientes actualmente 1,316 productos distintos, la selección de productos críticos se realizó en base a las ventas fundamentando la selección en el principio de Pareto, que especifica que pocos productos vitales generan la mayoría de las ventas y muchos productos triviales la minoría de las ventas, debido a limitaciones de tiempo y recursos como por ejemplo el software no se llegaron a seleccionar los 30 productos que corresponden al 80.38% de las ventas. Para el presente proyecto se han seleccionado 6 productos cuyas ventas del año 2017 totalizan \$551,354.68 y representan el 25,91% de las ventas del año 2017 cuyo valor asciende a \$2,128,050.90. A continuación, se muestran los 6 productos

seleccionados con información referente al código del producto, la descripción del producto, las ventas del año 2017 y el porcentaje que representan sus ventas con respecto al total de las ventas del año.

Item	Descripción del producto	Ventas año 2017	%
I1174	Sapolio Mata MoscaS 360ml	163,799.53	0.0770
I2793	Michu 18Kg Pollo	150,069.74	0.0705
I1175	Sapolio Mata Cucarachas 360ml	73,576.22	0.0346
I2245	Buen Can 30Kg ARMG Pollo	66,594.54	0.0313
I3567	Buen Can 30Kg CRMG Pollo	54,483.06	0.0256
I1429	Sapolio Deterg. 15kg Floral	42,831.58	0.0201
	Total	551,354.68	0.2591

Ilustración 1, “Selección de productos a simular”

### Distribución de probabilidad de los datos

El desarrollo de un modelo válido de los datos de entrada requiere los siguientes cuatro pasos básicos: 1) Recogida de datos del sistema real. 2) Identificación de la familia de distribución probabilística que representa el conjunto de datos de entrada, a partir de la distribución de frecuencia o histograma de los datos de entrada. 3) Estimar los parámetros que determinen una distribución específica de la familia de distribuciones identificada. 4) Test de validación de la distribución y los parámetros seleccionados.

Para poder realizar la simulación se procedió a determinar la distribución que siguen los datos: tiempo entre arribo de clientes, cantidad de producto que compra el cliente por factura, tiempo entre arribo de las entregas de los proveedores, cantidad de producto que entrega el proveedor por factura.

#### Distribución del tiempo entre arribo de clientes

No se dispone de datos del tiempo entre arribo de clientes sin embargo se realizó una aproximación con la información que se dispone, se sabe cuántas facturas de ventas se han realizado en el año y la contabilización de los días laborables del año (280) con lo que se obtuvo que la media de arribos por día es 10.04 clientes al día (8 horas), por lo tanto, por hora se atienden 1.255 clientes, es decir un cliente arriba en 0.80 horas que en minutos serían 47.80 minutos. Como distribución del tiempo de arribo entre clientes se ha supuesto una distribución exponencial con media de 47.80 minutos, el objeto del estudio no está orientado

a analizar las colas que se forman, lo que es realmente importante es simular el número de clientes que ingresa al sistema y determinar cuánto compra de cada producto pues es esto lo que afecta a las ventas que serán el referente para sugerir las compras.

#### Distribución de la cantidad de producto que compra el cliente por factura

Con el propósito de simular cuanto compran los clientes en cada factura por producto se procedió a contar la cantidad que se vendió en cada factura por producto, a continuación, se presenta por cada producto un análisis descriptivo de los datos, su diagrama de frecuencia, la distribución candidata con los parámetros estimados y la prueba de bondad de ajuste Chi Cuadrado de la distribución candidata que menor error cuadrado presenta, para estos análisis se utilizó el Input Analyzer de Arena.

- Producto 1 (I1174)

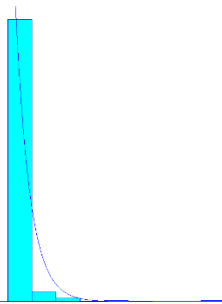


Ilustración 2, “Diagrama de la distribución del producto 1 (I1174)”

Distribution Summary	
Distribution:	Exponential
Expression:	-0.001 + EXPO(31.5)
Square Error:	0.049087
Chi Square Test	
Number of intervals	= 9
Degrees of freedom	= 6
Test Statistic	= 562
Corresponding p-value	< 0.005
Data Summary	
Number of Data Points	= 2811
Min Data Value	= 0
Max Data Value	= 4.2e+003
Sample Mean	= 31.5
Sample Std Dev	= 180
Histogram Summary	
Histogram Range	= -0.001 to 4.2e+003
Number of Intervals	= 40

Ilustración 3, “Cuadro resumen de estadísticos descriptivos del producto 1 (I1174)”

Como se puede observar en la prueba de Chi Cuadrado se rechaza  $H_0$ : Los datos siguen una distribución exponencial:  $-0.001 + \text{EXPO}(31.5)$ , ya que el p valor es menor a 0.05. Como se observa que los datos no se ajustan a ninguna distribución conocida se procede a utilizar la distribución empírica de la muestra en la simulación.

- Producto 2 (I2793)



Ilustración 4, “Diagrama de la distribución del producto 2 (I2793)”

Distribution Summary	
Distribution:	Exponential
Expression:	-0.001 + EXPO(1.81)
Square Error:	0.251195
Chi Square Test	
Number of intervals	= 13
Degrees of freedom	= 10
Test Statistic	= 2.48e+003
Corresponding p-value	< 0.005
Data Summary	
Number of Data Points	= 2811
Min Data Value	= 0
Max Data Value	= 100
Sample Mean	= 1.81
Sample Std Dev	= 5.36
Histogram Summary	
Histogram Range	= -0.001 to 100
Number of Intervals	= 40

Ilustración 5, “Cuadro resumen de estadísticos descriptivos del producto 2 (I2793)”

Como se puede observar en la prueba de Chi Cuadrado se rechaza  $H_0$ : Los datos siguen una distribución exponencial:  $-0.001 + \text{EXPO}(1.81)$ , ya que el p valor es menor a 0.05. Como se observa que los datos no se ajustan a ninguna distribución conocida se procede a utilizar la distribución empírica de la muestra en la simulación.

- Producto 3 (I1175)



Ilustración 6, “Diagrama de la distribución del producto 3 (I1175)”

Distribution Summary	
Distribution:	Exponential
Expression:	-0.001 + EXPO(14.1)
Square Error:	0.101099
Chi Square Test	
Number of intervals	= 8
Degrees of freedom	= 5
Test Statistic	= 956
Corresponding p-value	< 0.005
Data Summary	
Number of Data Points	= 2811
Min Data Value	= 0
Max Data Value	= 2.34e+003
Sample Mean	= 14.1
Sample Std Dev	= 77.9
Histogram Summary	
Histogram Range	= -0.001 to 2.34e+003
Number of Intervals	= 40

Ilustración 7, “Cuadro resumen de estadísticos descriptivos del producto 3 (I1175)”

Como se puede observar en la prueba de Chi Cuadrado se rechaza  $H_0$ : Los datos siguen una distribución exponencial:  $-0.001 + \text{EXPO}(14.1)$ , ya que el p valor es menor a 0.05. Como se observa que los datos no se ajustan a ninguna distribución conocida se procede a utilizar la distribución empírica de la muestra para la simulación.

- Producto 4 (I2245)

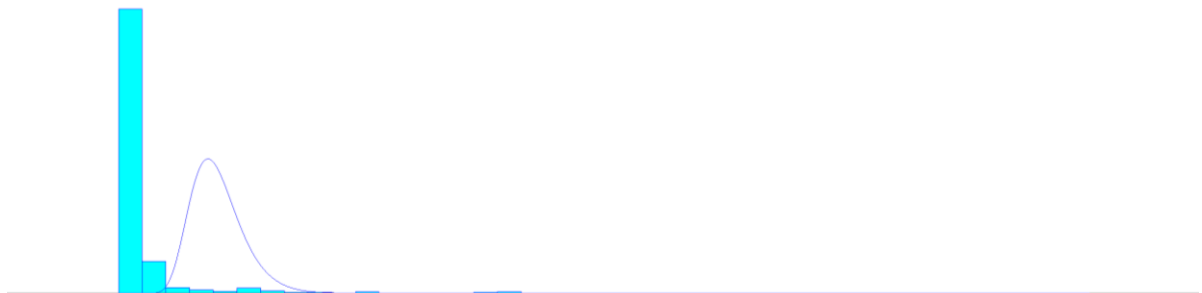


Ilustración 8, “Diagrama de la distribución del producto 4 (I2245)”

Distribution Summary	
Distribution:	Lognormal
Expression:	$-0.5 + \text{LOGN}(0.882, 0.741)$
Square Error:	0.458820
Chi Square Test	
Number of intervals	= 18
Degrees of freedom	= 15
Test Statistic	= 7.77e+003
Corresponding p-value	< 0.005
Data Summary	
Number of Data Points	= 2811
Min Data Value	= 0
Max Data Value	= 40
Sample Mean	= 0.62
Sample Std Dev	= 2.29
Histogram Summary	
Histogram Range	= -0.5 to 40.5
Number of Intervals	= 41

Ilustración 9, “Cuadro resumen de estadísticos descriptivos del producto 4 (I2245)”

Como se puede observar en la prueba de Chi Cuadrado se rechaza  $H_0$ : Los datos siguen una distribución lognormal:  $-0.5 + \text{LOGN}(0.882, 0.741)$ , ya que el p valor es menor a 0.05. Como se observa que los datos no se ajustan a ninguna distribución conocida se procede a utilizar la distribución empírica de la muestra en la simulación.

- Producto 5 (I3567)



Ilustración 10, “Diagrama de la distribución del producto 5 (I3567)”

Distribution Summary	
Distribution:	Beta
Expression:	$-0.5 + 17 * \text{BETA}(0.312, 5.39)$
Square Error:	0.004154
Chi Square Test	
Number of intervals	= 9
Degrees of freedom	= 6
Test Statistic	= 214
Corresponding p-value	< 0.005
Data Summary	
Number of Data Points	= 2811
Min Data Value	= 0
Max Data Value	= 16
Sample Mean	= 0.432
Sample Std Dev	= 1.49
Histogram Summary	
Histogram Range	= -0.5 to 16.5
Number of Intervals	= 17

Ilustración 11, “Cuadro resumen de estadísticos descriptivos del producto 5 (I3567)”

Como se puede observar en la prueba de Chi Cuadrado se rechaza  $H_0$ : Los datos siguen una distribución beta:  $-0.5 + 17 * \text{BETA}(0.315, 5.39)$ , ya que el p valor es menor a 0.05. Como se observa que los datos no se ajustan a ninguna distribución conocida se procede a utilizar la distribución empírica de la muestra

- Producto 6 (I1429)



Ilustración 12, “Diagrama de la distribución del producto 6 (I1429)”



Distribution Summary	
Distribution:	Lognormal
Expression:	-0.5 + LOGN(0.735, 0.53)
Square Error:	0.295987
Chi Square Test	
Number of intervals	= 23
Degrees of freedom	= 20
Test Statistic	= 3.1e+003
Corresponding p-value	< 0.005
Data Summary	
Number of Data Points	= 2811
Min Data Value	= 0
Max Data Value	= 75
Sample Mean	= 0.581
Sample Std Dev	= 3.64
Histogram Summary	
Histogram Range	= -0.5 to 75.5
Number of Intervals	= 76

Ilustración 13, “Cuadro resumen de estadísticos descriptivos del producto 6 (I1429)”

Como se puede observar en la prueba de Chi Cuadrado se rechaza  $H_0$ : Los datos siguen una distribución lognormal:  $-0.5 + \text{LOGN}(0.735, 0.53)$ , ya que el p valor es menor a 0.05. Como se observa que los datos no se ajustan a ninguna distribución conocida se procede a utilizar la distribución empírica de la muestra en la simulación.

#### Distribución del tiempo entre arribo de las entregas de proveedores

No se tiene datos del tiempo entre arribo de las entregas de los proveedores, pero se cuenta con la información de cuantas facturas de compra se han realizado en el año y el total de los días laborables del año (280) con lo que se obtuvo la media de arribos por día que es 0.25 proveedores al día (8 horas), por lo tanto, un proveedor arriba en 32 horas. Como distribución del tiempo de arribo entre proveedores se ha utilizado una distribución exponencial con media 32 horas, considerando que el objeto del estudio no está orientado a analizar las colas que se forman y que lo relevante es simular el número de pedidos realizados a los proveedores que ingresa al sistema y determinar cuánto se compra de cada producto ya que es esto lo que afecta el inventario.

## Distribución de la cantidad de producto que entrega el proveedor por factura

Con la finalidad de simular cuanto se compra en cada factura por producto se procedió a contar la cantidad por producto que se compró en cada factura. A continuación, se muestra por cada producto un análisis descriptivo de los datos, su diagrama de frecuencias, la distribución candidata con los parámetros estimados y la prueba de bondad de ajuste Chi Cuadrado de la distribución candidata que menor error cuadrado presenta, para estos análisis se utilizó el Input Analyzer de Arena.

- Producto 1 (I1174)

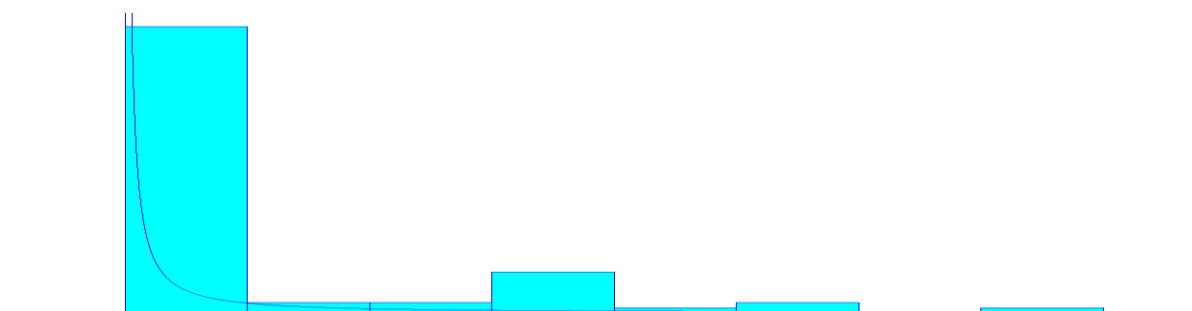


Ilustración 14, “Diagrama de la distribución del producto 1 (I1174)”

Distribution Summary	
Distribution:	Weibull
Expression:	-0.001 + WEIB(1.34, 0.136)
Square Error:	0.034423
Chi Square Test	
Number of intervals	= 1
Degrees of freedom	= -2
Test Statistic	= 1.78
Corresponding p-value	< 0.005
Data Summary	
Number of Data Points	= 72
Min Data Value	= 0
Max Data Value	= 1.2e+004
Sample Mean	= 1.24e+003
Sample Std Dev	= 2.62e+003
Histogram Summary	
Histogram Range	= -0.001 to 1.2e+004
Number of Intervals	= 8

Ilustración 15, “Cuadro resumen de estadísticos descriptivos del producto 1 (I1174)”

Como se puede observar en la prueba de Chi Cuadrado se rechaza  $H_0$ : Los datos siguen una distribución weibull:  $-0.001+WEIB(1.34,0.136)$ , ya que el p valor es menor a 0.05. Como

se observa que los datos no se ajustan a ninguna distribución conocida se procede a utilizar la distribución empírica de la muestra en la simulación.

- Producto 2 (I2793)

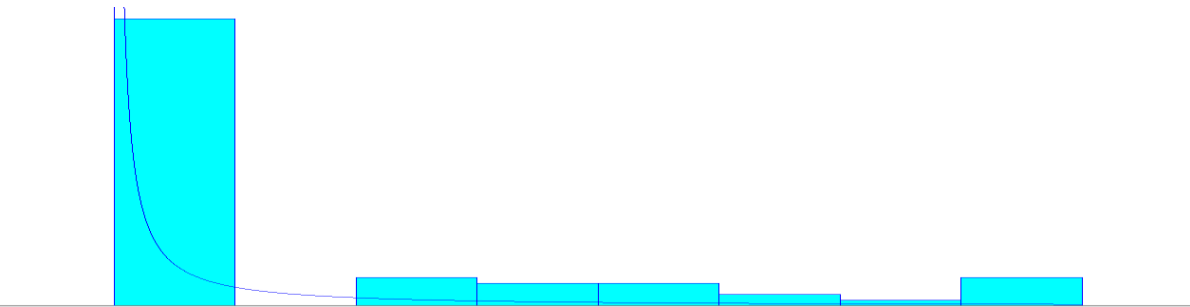


Ilustración 16, “Diagrama de la distribución del producto 2 (I2793)”

Distribution Summary	
Distribution:	Weibull
Expression:	-0.001 + WEIB(0.825, 0.171)
Square Error:	0.038473
Chi Square Test	
Number of intervals	= 1
Degrees of freedom	= -2
Test Statistic	= 2.34
Corresponding p-value	< 0.005
Data Summary	
Number of Data Points	= 72
Min Data Value	= 0
Max Data Value	= 405
Sample Mean	= 70.4
Sample Std Dev	= 122
Histogram Summary	
Histogram Range	= -0.001 to 405
Number of Intervals	= 8

Ilustración 17, “Cuadro resumen de estadísticos descriptivos del producto 2 (I2793)”

Como se puede observar en la prueba de Chi Cuadrado se rechaza  $H_0$ : Los datos siguen una distribución weibull:  $-0.001 + WEIB(0.825, 0.171)$  ya que el p valor es menor a 0.05. Como se observa que los datos no se ajustan a ninguna distribución conocida se procede a utilizar la distribución empírica de la muestra en la simulación.

- Producto 3 (I1175)

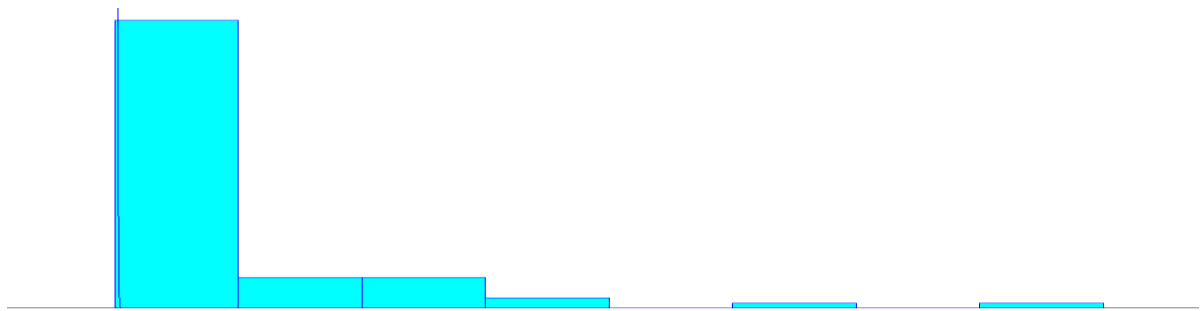


Ilustración 18, “Diagrama de la distribución del producto 3 (I1175)”

Distribution Summary	
Distribution:	Exponential
Expression:	-0.001 + EXPO(44.8)
Square Error:	0.033221
Chi Square Test	
Number of intervals	= 1
Degrees of freedom	= -2
Test Statistic	= 1.82
Corresponding p-value	< 0.005
Data Summary	
Number of Data Points	= 72
Min Data Value	= 0
Max Data Value	= 600
Sample Mean	= 44.8
Sample Std Dev	= 106
Histogram Summary	
Histogram Range	= -0.001 to 600
Number of Intervals	= 8

Ilustración 19, “Cuadro resumen de estadísticos descriptivos del producto 3 (I1175)”

Como se puede observar en la prueba de Chi Cuadrado se rechaza  $H_0$ : Los datos siguen una distribución exponencial:  $-0.001 + \text{EXPO}(44.8)$  ya que el p valor es menor a 0.05. Como se observa que los datos no se ajustan a ninguna distribución conocida se procede a utilizar la distribución empírica de la muestra en la simulación.

- Producto 4 (I2245)

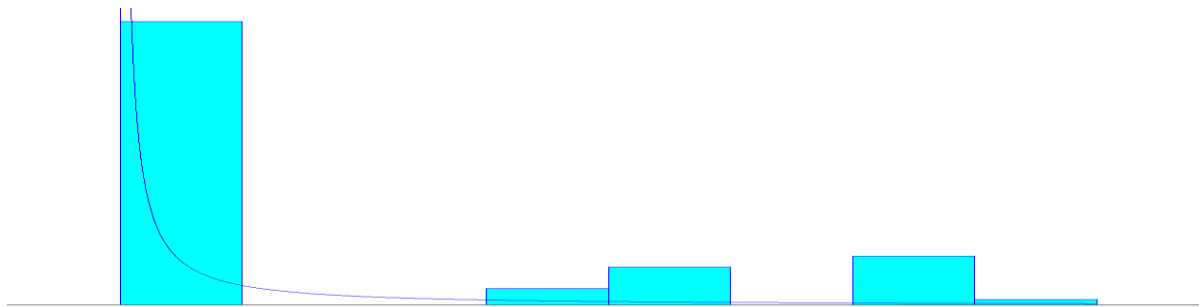


Ilustración 20, “Diagrama de la distribución del producto 4 (I2245)”

Distribution Summary	
Distribution:	Weibull
Expression:	-0.001 + WEIB(0.364, 0.184)
Square Error:	0.045200
Chi Square Test	
Number of intervals	= 1
Degrees of freedom	= -2
Test Statistic	= 1.91
Corresponding p-value	< 0.005
Data Summary	
Number of Data Points	= 72
Min Data Value	= 0
Max Data Value	= 126
Sample Mean	= 23.9
Sample Std Dev	= 40.4
Histogram Summary	
Histogram Range	= -0.001 to 126
Number of Intervals	= 8

Ilustración 21, “Cuadro resumen de estadísticos descriptivos del producto 4 (I2245)”

Como se puede observar en la prueba de Chi Cuadrado se rechaza  $H_0$ : Los datos siguen una distribución weibull:  $-0.001 + WEIB(0.364, 0.184)$  ya que el p valor es menor a 0.05. Como se observa que los datos no se ajustan a ninguna distribución conocida se procede a utilizar la distribución empírica de la muestra en la simulación.

- Producto 5 (I3567)

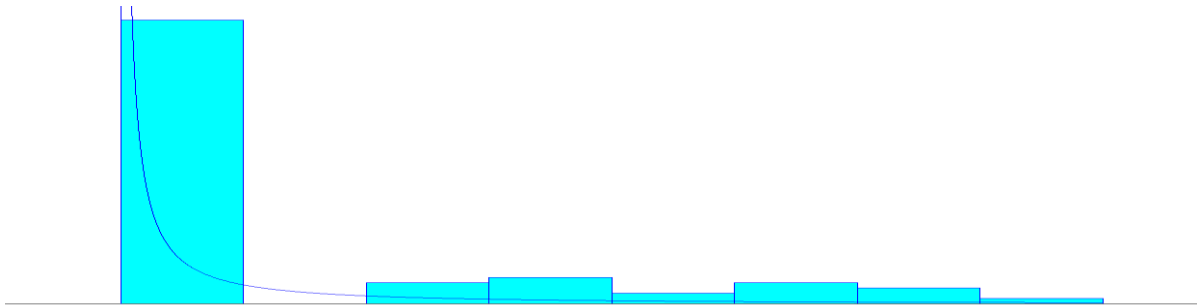


Ilustración 22, “Diagrama de la distribución del producto 5 (I3567)”

Distribution Summary	
Distribution:	Weibull
Expression:	-0.001 + WEIB(0.29, 0.191)
Square Error:	0.030119
Chi Square Test	
Number of intervals	= 1
Degrees of freedom	= -2
Test Statistic	= 1.77
Corresponding p-value	< 0.005
Data Summary	
Number of Data Points	= 72
Min Data Value	= 0
Max Data Value	= 110
Sample Mean	= 16.6
Sample Std Dev	= 30.2
Histogram Summary	
Histogram Range	= -0.001 to 110
Number of Intervals	= 8

Ilustración 23, “Cuadro resumen de estadísticos descriptivos del producto 5 (I3567)”

Como se puede observar en la prueba de Chi Cuadrado se rechaza  $H_0$ : Los datos siguen una distribución weibull:  $-0.001 + WEIB(0.29, 0.19)$  ya que el p valor es menor a 0.05. Como se observa que los datos no se ajustan a ninguna distribución conocida se procede a utilizar la distribución empírica de la muestra en la simulación.

- Producto 6 (I1429)

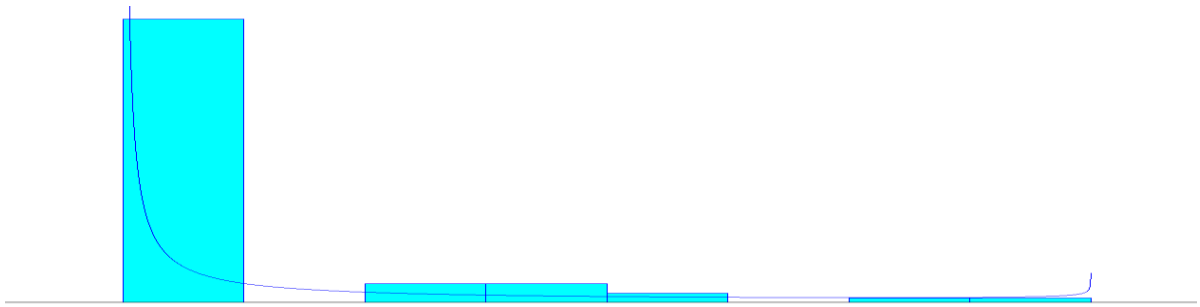


Ilustración 24, “Diagrama de la distribución del producto 6 (I1429)”

Distribution Summary	
Distribution:	Beta
Expression:	$-0.001 + 350 * \text{BETA}(0.0652, 0.736)$
Square Error:	0.007865
Chi Square Test	
Number of intervals	= 2
Degrees of freedom	= -1
Test Statistic	= 1.33
Corresponding p-value	< 0.005
Data Summary	
Number of Data Points	= 72
Min Data Value	= 0
Max Data Value	= 350
Sample Mean	= 28.5
Sample Std Dev	= 71.3
Histogram Summary	
Histogram Range	= -0.001 to 350
Number of Intervals	= 8

Ilustración 25, “Cuadro resumen de estadísticos descriptivos del producto 6 (I1429)”

Como se puede observar en la prueba de Chi Cuadrado se rechaza  $H_0$ : Los datos siguen una distribución beta:  $-0.001 + 350 * \text{BETA}(0.0652, 0.736)$  ya que el p valor es menor a 0.05. Como se observa que los datos no se ajustan a ninguna distribución conocida se procede a utilizar la distribución empírica de la muestra en la simulación.

## Sistema Real

El proceso de compra y despacho de productos inicia cuando el cliente arriba al almacén si el vendedor está disponible es atendido inmediatamente, si por el contrario el vendedor está ocupado atendiendo a otro cliente se genera una cola, la cola es atendida por el vendedor

mediante el método FIFO (first in first out). El vendedor es el encargado de atender y proporcionarle al cliente toda la información que solicite como: consulta de precios, especificación de los productos, etc. Cuando el cliente desea realizar un pedido de un producto el vendedor procede a facturar el producto en caso de que no se disponga del inventario el sistema muestra un mensaje de error indicando que no es posible facturar tal cantidad y muestra la cantidad que se dispone en el inventario actual, el vendedor procede a consultarle al cliente si desea comprar las unidades disponibles en el inventario, si el cliente no acepta se pierde la venta, caso contrario se facturan las unidades disponibles en el inventario.

Una vez ya realizado el pedido el sistema envía una orden de despacho a bodega. El bodeguero es el encargado de despachar los pedidos de los clientes. Si el bodeguero está disponible el pedido es despachado inmediatamente, si por el contrario el bodeguero está ocupado despachando otro pedido se genera una cola, la cola de clientes es atendida por el bodeguero mediante el método FIFO (first in first out), por el contrario si en la cola hay proveedores ellos tendrían la preferencia y de igual forma se atendería a los proveedores mediante el método FIFO. Cuando el cliente recibe su pedido se retira del almacén.

Actualmente el proceso de compras dentro de la empresa se lleva a cabo de forma empírica, se revisa diariamente al final del día los inventarios y según el criterio del jefe de compras se pide una determinada cantidad, esta cantidad no responde a ningún criterio de manejo de inventarios en específico salvo la experiencia del jefe de compras, es decir no se tiene definido un punto de reorden por producto, ni una periodicidad determinada para realizar las compras de productos. Cada vez que el jefe de compras realiza un pedido al proveedor genera una orden de compra en el sistema.

El bodeguero es también el encargado de recibir las entregas de los productos por parte de los proveedores y ubicarlos de forma ordenada en las estanterías, luego de constatar las cantidades de productos recibidas, el bodeguero procede a aceptar la orden de compra en el sistema esto genera que el inventario se actualice considerando las cantidades recibidas por parte del proveedor.

## **SDL**

SDL, Specification and Description Language por sus siglas en inglés, es un lenguaje estándar y gráfico de especificación y descripción para modelos simulación discreta, que busca definir



el sistema de forma sencilla a fin de que sea comprensible para todos los involucrados en el diseño de la simulación. El lenguaje está orientado a eventos, es decir describe el comportamiento del proceso por medio de la ocurrencia de eventos, los eventos son aquellos que realizan un cambio en el estado del sistema, en SDL se conocen como señales, las señales se mueven a través del sistema mediante los canales. El lenguaje tiene 4 niveles: 1) sistema que contiene bloques, 2) bloques que contienen procesos u otros bloques, 3) procesos, que es un agente que describe el comportamiento y 4) procedimientos que encapsulan partes de los procesos, los 4 de niveles del modelo se detallan en la siguiente sección.

## Modelo en SDL sistema real

### Sistema COFEGO

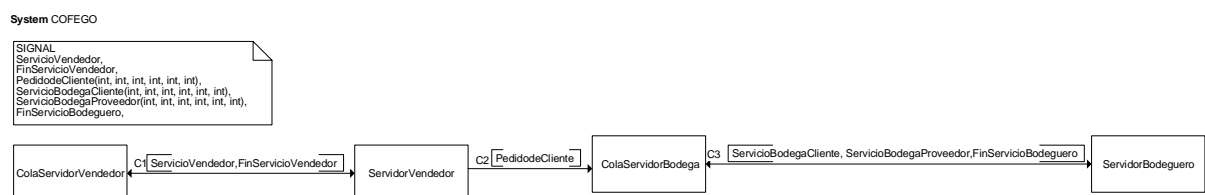


Ilustración 26, “Sistema COFEGO”

Se puede observar que el sistema está compuesto por 4 bloques: ColaServidorVendedor, ServidorVendedor, ColaServidorBodega, ServidorBodeguero, adicionalmente se puede apreciar que los eventos, representados mediante señales en SDL son: ServicioVendedor, FinServicioVendedor, Pedidodecliente, ServicioBodegaCliente, ServicioBodegaProveedor, FinServicioBodeguero; estas señales son transportadas mediante los canales: C1, C2 y C3. También se realiza una definición de las señales del sistema donde se especifica el nombre de la señal y los parámetros que incluye, se especifica también el tipo de parámetro, en este caso todos son enteros.

## Bloque Cola Servidor Vendedor

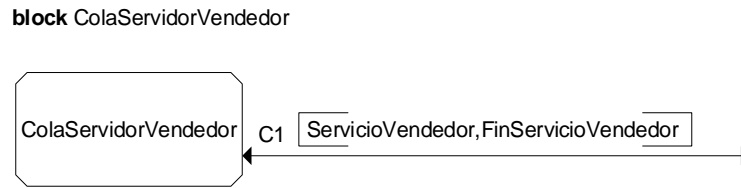


Ilustración 27, “Bloque Cola Servidor Vendedor”

El Bloque ColaServidorVendedor está compuesto por un único proceso ColaServidorVendedor el cual envía la señal ServicioVendedor y recibe la señal FinServicioVendedor del Bloque ServidorVendedor a través del canal C1.

## Bloque Servidor Vendedor

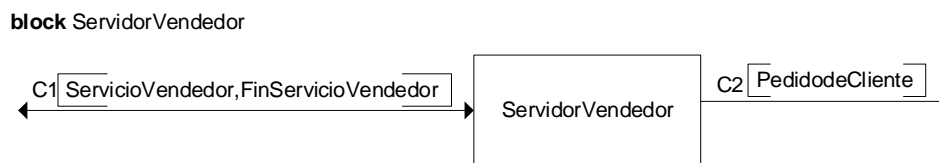


Ilustración 28, “Bloque Servidor Vendedor”

El Bloque ServidorVendedor está compuesto por un único proceso ServidorVendedor el cual envía la señal FinServicio Vendedor y recibe la señal ServicioVendedor del Bloque ColaServidorBodega a través del canal C1, también envía la señal PedidodeCliente al Bloque ColaServidorBodega mediante el canal C2.

## Bloque Cola Servidor Bodega



Ilustración 29, “Bloque Cola Servidor Bodega”

El Bloque ColaServidorBodega está compuesto por un único proceso ColaServidorBodega el cual recibe la señal PedidodeCliente del Bloque ServidorVendedor a través del canal C2,

también envía las señales ServicioBodegaCliente y ServicioBodegaProveedor, y recibe la señal FinServicioBodeguero del Bloque ServidorBodeguero mediante el canal C3.

**Bloque Servidor Bodeguero**



Ilustración 30, “Bloque Cola Servidor Bodeguero”

El Bloque ServidorBodeguero está compuesto por un único proceso ServidorBodeguero el cual envía la señal FinServicioBodeguero y recibe las señales ServicioBodegaCliente y ServicioBodegaProveedor del Bloque ColaServidorBodega mediante el canal C3.

# Proceso Cola Servidor Vendedor

Process: ColaServidorVendedor

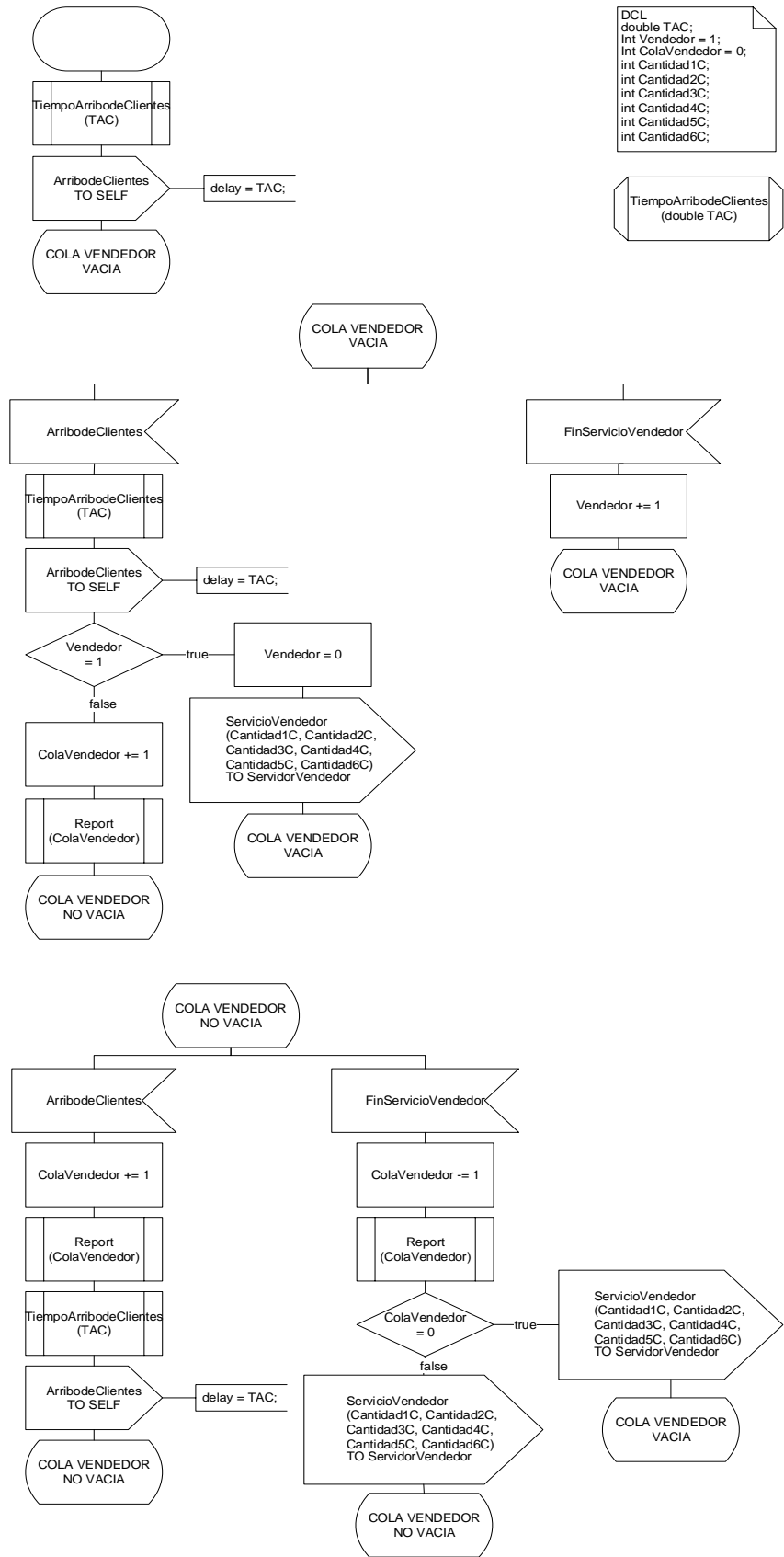


Ilustración 31, "Proceso Cola Servidor Vendedor"

La cola del servidor vendedor es representada mediante los estados “COLA VENDEDOR VACÍA” y “COLA VENDEDOR NO VACÍA”, luego del inicio el proceso refleja el arribo del primer cliente, el tiempo arribo entre clientes se calcula mediante el procedimiento “TiempoArribodeClientes”, luego se envía la señal “ArribodeClientes” al mismo proceso, la señal tiene un retraso acorde al parámetro “TAC” calculado mediante el procedimiento, el estado inicial es “COLA VENDEDOR VACÍA”.

En el estado “COLA VENDEDOR VACÍA” pueden ocurrir dos tipos de eventos representados mediante las señales: “ArribodeClientes” y “FinServicioVendedor”. Los mismos que se detallan a continuación:

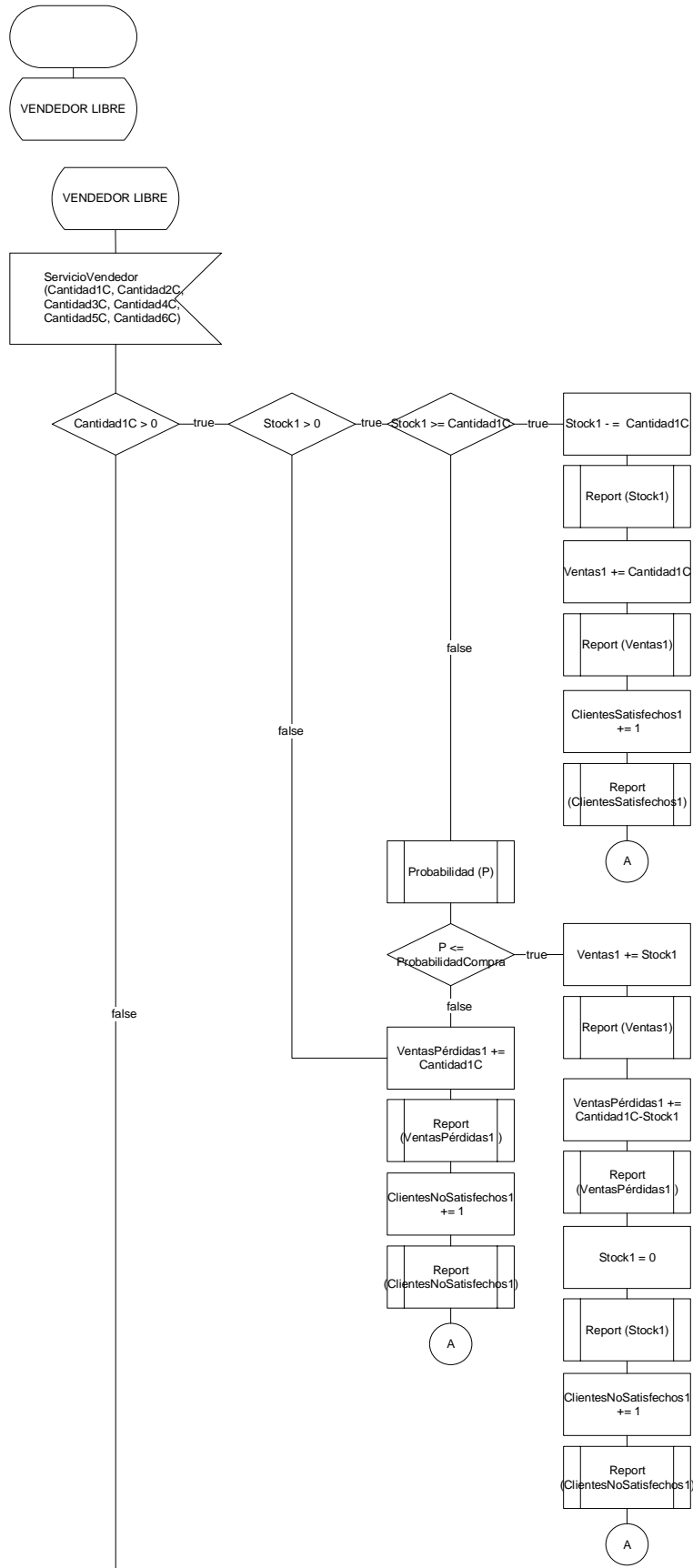
- Cuando el sistema con estado “COLA VENDEDOR VACÍA” recibe la señal “ArribodeClientes” debe generar el arribo del siguiente cliente para lo cual debe calcular el tiempo de arribo entre clientes mediante el procedimiento “TiempoArribodeClientes” que devuelve el parámetro “TAC” el mismo que es el tiempo entre arribos, luego envía nuevamente la señal “ArribodeClientes” al mismo proceso con un retraso especificado en el parámetro “TAC”, denotando así la generación del siguiente cliente. Luego se consulta si el vendedor está disponible (Vendedor= 1), el vendedor está representado por la variable Vendedor, cuyo valor al inicio de la simulación es de 1 indicando que está disponible, en caso de que el vendedor esté disponible, se cambia el valor de la variable Vendedor a cero (Vendedor= 0), para indicar que ya no está disponible y se envía la señal “ServicioVendedor” al proceso “ServidorVendedor” para que el vendedor brinde el servicio, la señal contiene 6 parámetros que corresponden al número de unidades que el cliente desea comprar de cada uno de los 6 productos seleccionados para la simulación, el sistema se mantiene en el estado “COLA VENDEDOR VACÍA”. En caso de que el vendedor no esté disponible (Vendedor= 0) se aumenta la cola en una unidad (ColaVendedor += 1), luego se solicita un reporte de la cola y se cambia el estado a “COLA VENDEDOR NO VACÍA”.
- Cuando el sistema con estado “COLA VENDEDOR VACÍA” recibe la señal “FinServicioVendedor”, se indica que el servidor vendedor ha sido liberado por lo que se cambia el valor de la variable Vendedor a uno (Vendedor= 1) y se mantiene el estado del sistema en “COLA VENDEDOR VACÍA”.

En el estado “COLA VENDEDOR NO VACÍA” pueden ocurrir dos tipos de eventos representados mediante señales: “ArribodeClientes” y “FinServicioVendedor”. Los mismos que se detallan a continuación:

- Cuando el sistema con estado “COLA VENDEDOR NO VACÍA” recibe la señal “ArribodeClientes” se aumenta la cola en una unidad ( $\text{ColaVendedor} += 1$ ), luego se solicita un reporte de la cola, posteriormente se calcula el tiempo entre arribos para el siguiente cliente por medio del procedimiento “TiempoArribodeClientes” que devuelve el parámetro “TAC”, después se envía la señal “ArribodeClientes” al mismo proceso con un retraso determinado por el parámetro “TAC”, de esta forma se genera el siguiente cliente y se mantiene el estado en “COLA VENDEDOR NO VACÍA”.
- Cuando el sistema con estado “COLA VENDEDOR NO VACÍA” recibe la señal “FinServicioVendedor”, se disminuye la cola en una unidad ( $\text{ColaVendedor} -= 1$ ), ya que se atendería a uno de los clientes de la cola, luego se solicita un reporte de la cola y se consulta si la cola ha quedado vacía ( $\text{ColaVendedor} = 0$ ); si la cola ha quedado vacía se envía la señal “ServicioVendedor” para que el vendedor brinde el servicio y se cambia el estado a “COLA VENDEDOR VACÍA”, si por el contrario la cola no ha quedado vacía ( $\text{ColaVendedor} \geq 1$ ) se envía la señal “ServicioVendedor” para que el vendedor brinde el servicio y se mantiene el estado en “COLA VENDEDOR NO VACÍA”.

# Proceso Servidor Vendedor

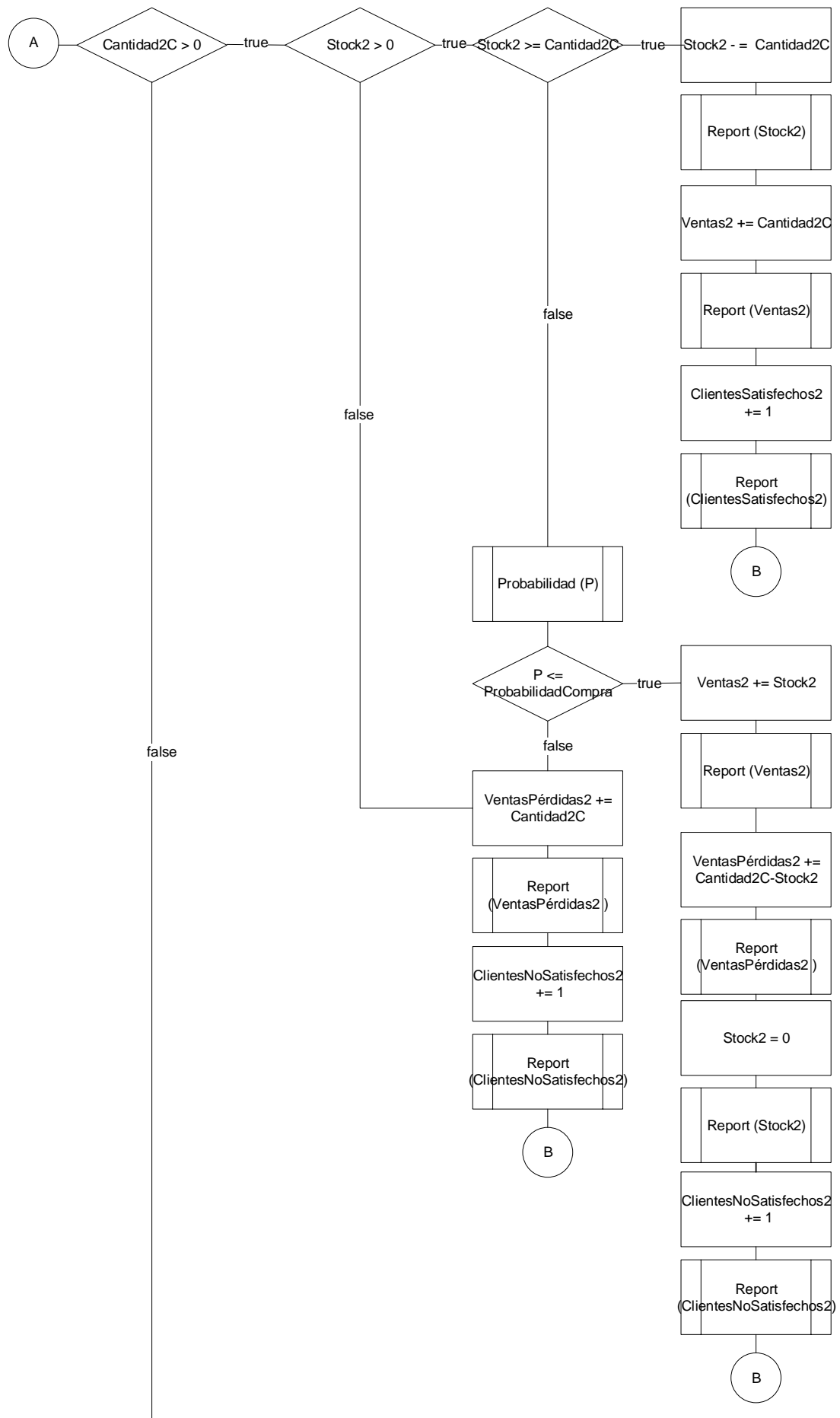
Process: ServidorVendedor



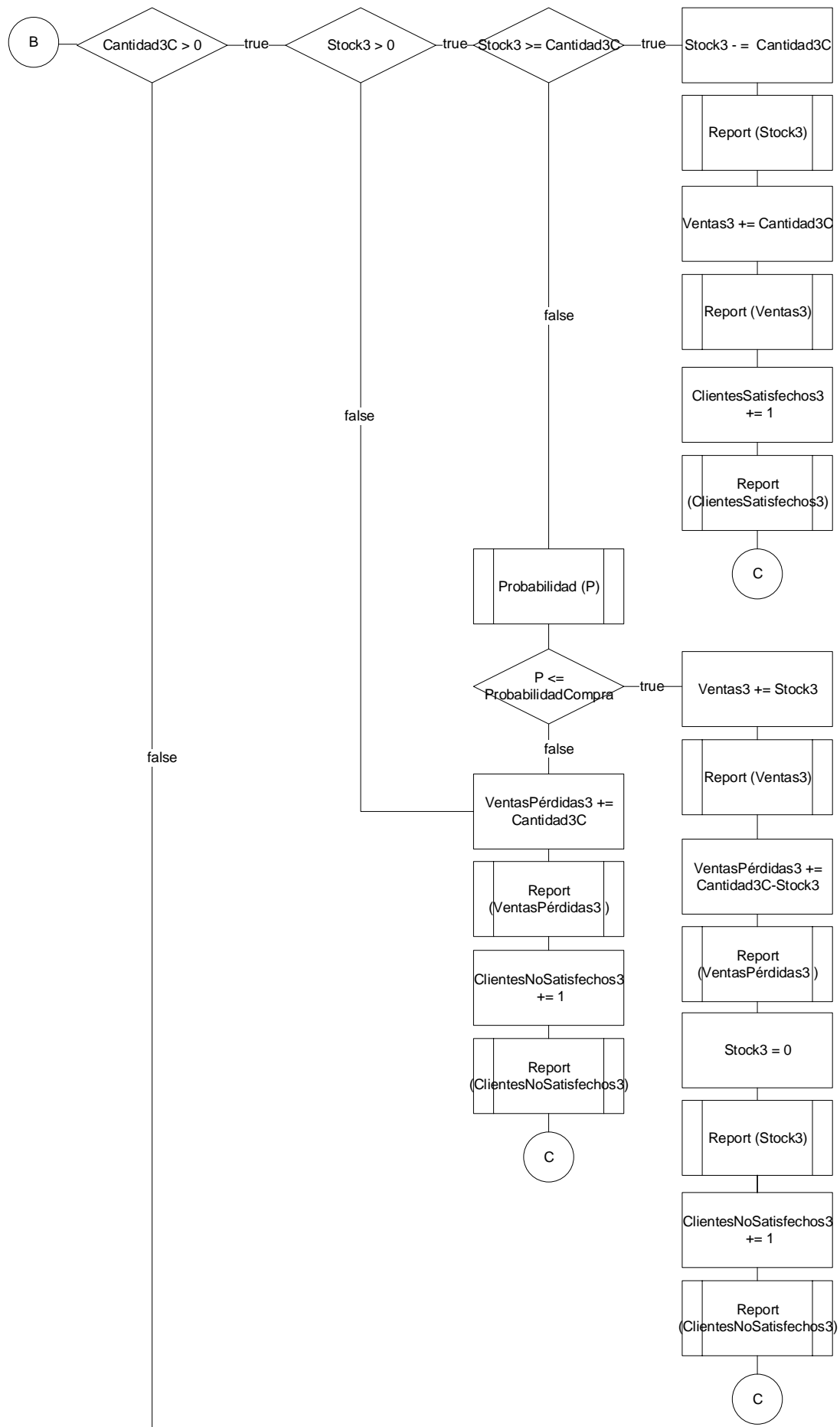
DCL  
double TSV;  
double P;  
double ProbabilidadCompra = 0.98;  
int Cantidad1C;  
int Cantidad2C;  
int Cantidad3C;  
int Cantidad4C;  
int Cantidad5C;  
int Cantidad6C;  
int Stock1 = 468;  
int Stock2 = 60;  
int Stock3 = 372;  
int Stock4 = 12;  
int Stock5 = 29;  
int Stock6 = 46;  
int VentasPérdidas1 = 0;  
int VentasPérdidas2 = 0;  
int VentasPérdidas3 = 0;  
int VentasPérdidas4 = 0;  
int VentasPérdidas5 = 0;  
int VentasPérdidas6 = 0;  
int Ventas1 = 0;  
int Ventas2 = 0;  
int Ventas3 = 0;  
int Ventas4 = 0;  
int Ventas5 = 0;  
int Ventas6 = 0;  
int ClientesSatisfechos1 = 0;  
int ClientesSatisfechos2 = 0;  
int ClientesSatisfechos3 = 0;  
int ClientesSatisfechos4 = 0;  
int ClientesSatisfechos5 = 0;  
int ClientesSatisfechos6 = 0;  
int ClientesNoSatisfechos1 = 0;  
int ClientesNoSatisfechos2 = 0;  
int ClientesNoSatisfechos3 = 0;  
int ClientesNoSatisfechos4 = 0;  
int ClientesNoSatisfechos5 = 0;  
int ClientesNoSatisfechos6 = 0;  
int ClientesAtendidosVendedor = 0;

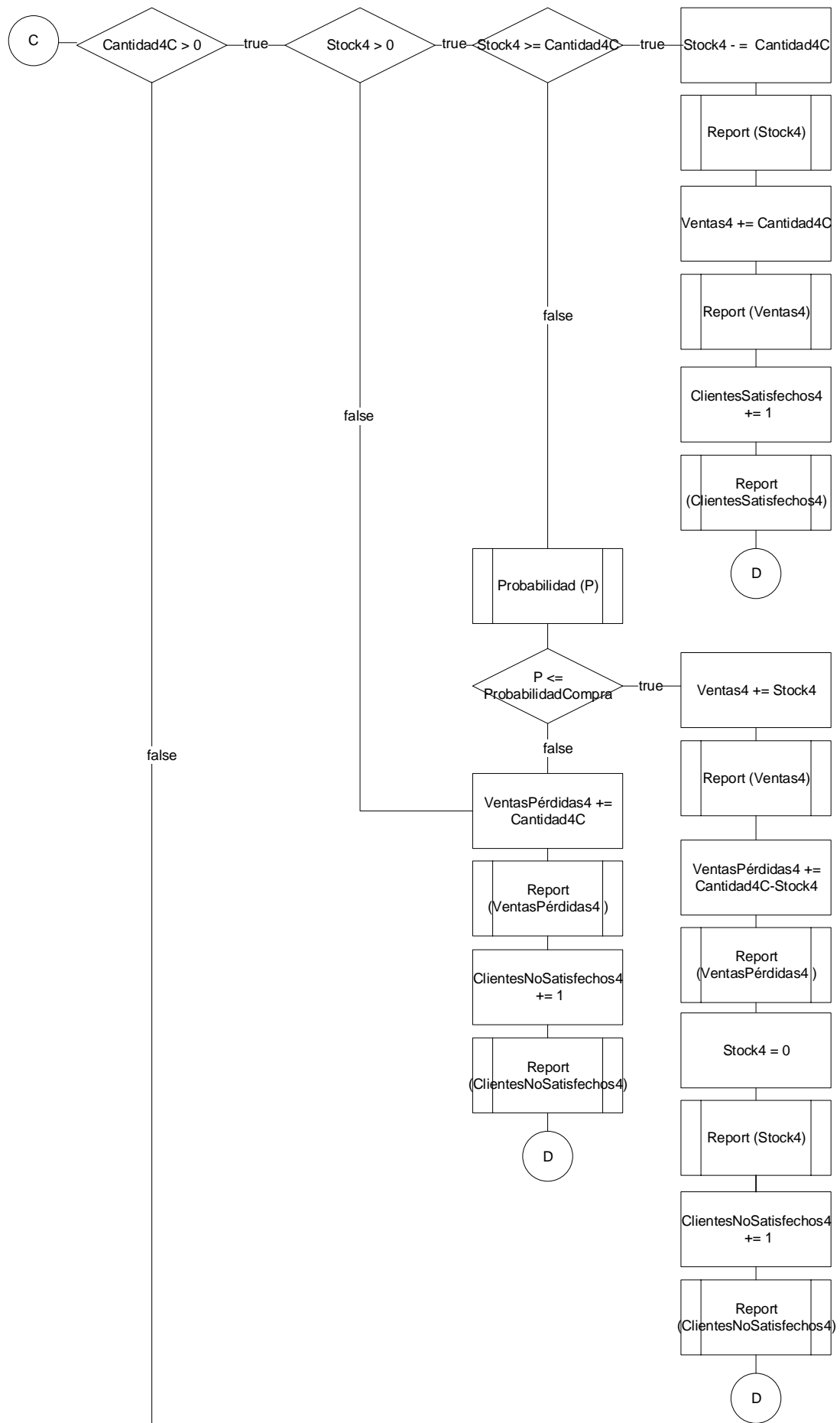
TiempoServicioVendedor  
(double TSV)

Probabilidad  
(double P)









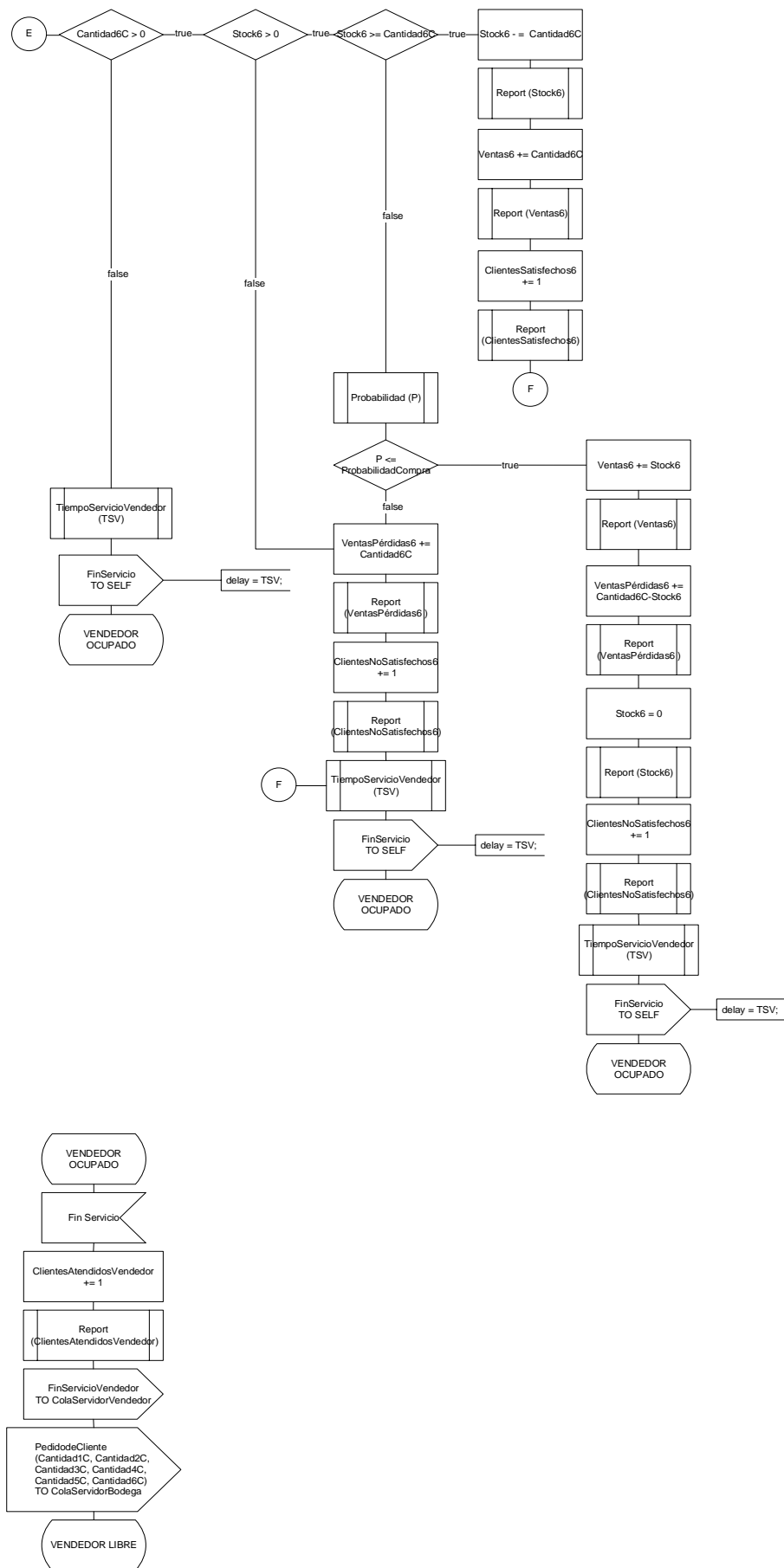


Ilustración 32, “Proceso Servidor Vendedor”

El proceso servidor vendedor es representado mediante los estados “VENDEDOR LIBRE” y “VENDEDOR OCUPADO”, el estado inicial del proceso es “VENDEDOR LIBRE”.

En el estado “VENEDOR LIBRE” puede ocurrir 1 tipo de evento representado mediante la señal “ServicioVendedor”. A continuación, se describe el proceso:

- El estado “VENDEDOR LIBRE” recibe la señal “ServicioVendedor” que contiene 6 parámetros los mismos que corresponden a las cantidades que desea comprar el cliente de los 6 productos a simular, para la explicación del diagrama tomaremos como referencia el producto 1. Se requiere preguntar si el cliente desea comprar el producto 1 ( $Cantidad1C > 0$ ) si es cierto, posteriormente se consulta si existe inventario del producto 1 ( $Stock1 > 0$ ), si es verdadero se pregunta si la variable del inventario del producto que el cliente desea comprar es mayor o igual que la cantidad que el cliente desea comprar del producto 1 ( $Stock1 \geq Cantidad1C$ ), en caso de ser afirmativa la respuesta, se procede a disminuir la variable de inventario restándole la cantidad que desea comprar el cliente ( $Stock1 -= Cantidad1C$ ), luego se solicita un reporte del inventario, también se incrementan las ventas del producto en la cantidad de unidades compradas por el cliente ( $Ventas1 += Cantidad1C$ ), se pide un reporte de ventas y se procede a aumentar el contador de clientes satisfechos ( $ClientesSatisfechos1 += 1$ ), contador que se utilizará para calcular posteriormente el nivel de servicio, se solicita un reporte del contador y se observa que el proceso se enlaza con el conector “A”. Por el contrario, en el caso que no exista inventario del producto ( $Stock1 = 0$ ), se procede a contar las cantidades del producto solicitadas por el cliente como ventas perdidas ( $VentasPerdidas1 += Cantidad1C$ ), ventas perdidas es la variable que acumula la cantidad de productos que por falta de inventario no se pudieron vender, inicialmente en la simulación la variable de ventas perdidas tiene un valor de 0, posteriormente se solicita un reporte de la variable ventas perdidas, adicionalmente se incrementa el contador de clientes no satisfechos ( $ClientesNoSatisfechos1 += 1$ ), se solicita un reporte de dicho contador y se visualiza que el proceso se enlaza con el conector “A”. Cuando el inventario del producto no sea mayor o igual a lo que el cliente desea comprar ( $Stock1 < Cantidad1C$ ), mediante el procedimiento “Probabilidad” que devuelve un parámetro “P”, el cual es un número aleatorio entre cero y cien, se procede a preguntar si “P” es menor o igual a la variable “ProbabilidadCompra” ( $P \leq ProbabilidadCompra$ ), variable que en la simulación se ha fijado en 0.98, en caso de ser cierto significa que el cliente decide comprar las unidades del producto disponibles en el inventario actual por lo que incrementamos las ventas en la cantidad compradas

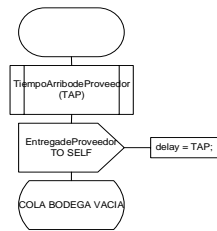
por el cliente que es equivalente al inventario del producto ( $Ventas1 += Stock1$ ), reportamos las ventas, a continuación se contabilizan las unidades no vendidas por falta de inventario como ventas perdidas ( $VentasPerdidas1 += Cantidad1C - Stock1$ ), se reportan las ventas perdidas, fijamos a cero el inventario del producto ( $Stock1=0$ ) puesto que todas las unidades se vendieron, reportamos el inventario del producto, posteriormente se aumenta el contador de “ClientesNoSatisfechos1” en una unidad, se reporta la variable “ClientesNoSatisfechos1” y se observa que el proceso se enlaza con el conector “A”. En caso de que el cliente decida no comprar las unidades del producto disponibles en el inventario ( $P > ProbabilidadCompra$ ), se procede a calcular las ventas perdidas ( $VentasPerdidas1 += Cantidad1C$ ), posteriormente se solicita un reporte de la variable ventas perdidas, seguidamente se incrementa el contador de clientes no satisfechos ( $ClientesNoSatisfechos1 += 1$ ), se reportan los clientes no satisfechos y se visualiza que el proceso se enlaza con el conector “A”. El conector “A” se enlaza con la pregunta si el cliente desea comprar el producto 2 ( $Cantidad2C > 0$ ), a partir de esta decisión el proceso para el producto 2 es igual que el anteriormente descrito para el producto 1, y posteriormente se realizará lo mismo con cada producto de la simulación. Luego, se observa que se calcula el tiempo de servicio del vendedor, mediante el procedimiento “TiempoServicioVendedor” que devuelve el parámetro “TSV” el mismo que es el tiempo de servicio, luego envía al mismo proceso la señal “FinServicio” que tendrá un retraso del parámetro “TSV” y durante ese tiempo se mantendría el estado “VENDEDOR OCUPADO”.

En el estado “VENDEDOR OCUPADO” puede ocurrir 1 tipo de evento representado mediante la señal “FinServicio”. A continuación, se describe el proceso:

- En el estado “VENDEDOR OCUPADO” se recibe la señal “FinServicio”, luego se van a contar los clientes atendidos por el vendedor con la variable “ClientesAtendidosVendedor” la misma que al inicio de la simulación se inicializa con cero, luego se reporta el total de los “ClientesAtendidosVendedor”, posteriormente se envía la señal “FinServicioVendedor” a la Cola Servidor Vendedor, notificando que el servidor está desocupado y ha terminado el servicio, adicionalmente también se envía la señal “PedidodeCliente” a la Cola Servidor Bodega y el estado del sistema cambia a “VENDEDOR LIBRE”.

## Proceso Cola Servidor Bodega

Process: ColaServidorBodega



DCL  
double TAP;  
int Cantidad1C;  
int Cantidad2C;  
int Cantidad3C;  
int Cantidad4C;  
int Cantidad5C;  
int Cantidad6C;  
Int ProveedorenCola = 0;  
Int Bodeguero = 0;  
Int ColaBodega = 0;  
Int Cantidad1P;  
Int Cantidad2P;  
Int Cantidad3P;  
Int Cantidad4P;  
Int Cantidad5P;  
Int Cantidad6P;

TiempoArribodeProveedor  
(double TAP)

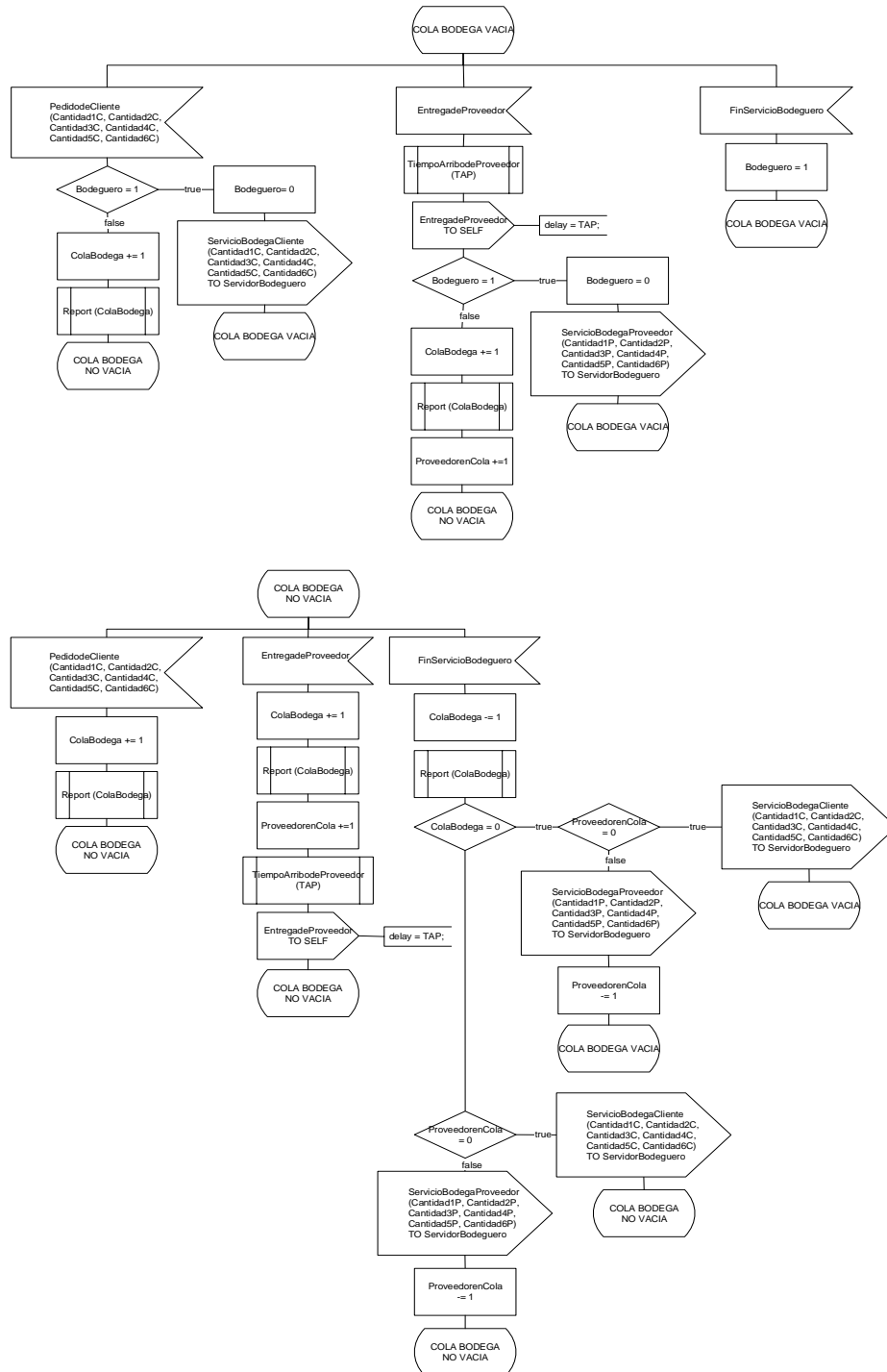


Ilustración 33, "Proceso Cola Servidor Bodega"

La cola del servidor bodega es representada mediante los estados “COLA BODEGA VACÍA” y “COLA BODEGAA NO VACÍA”, luego del inicio el proceso refleja el arribo del primer proveedor que entrega un pedido realizado por COFEGO, el tiempo arribo entre proveedores se calcula mediante un procedimiento “TiempoArribodeProveedor”, luego se envía la señal “EntregadeProveedor” al mismo proceso, la señal tiene un retraso acorde al parámetro “TAP” que es el tiempo calculado mediante el procedimiento, de esta forma se genera el primer arribo de los proveedores, el estado inicial es “COLA BODEGA VACÍA”.

En el estado “COLA BODEGA VACÍA” pueden ocurrir 3 tipos de eventos representados mediante señales: “PedidodeCliente”, “EntregadeProveedor” y “FinServicioBodeguero”. Los mismos que se detallan a continuación:

- El sistema con estado “COLA BODEGA VACÍA” recibe la señal “PedidodeCliente” posteriormente se debe preguntar si el Servidor Bodeguero está disponible (Bodeguero = 1), si esto es verdadero puesto que consecutivamente se enviará un servicio al servidor debemos indicar que el servidor no está disponible (Bodeguero = 0), a continuación, se envía la señal “ServicioBodegaCliente” al Servidor Bodeguero, el estado del sistema se mantendrá como “COLA BODEGA VACÍA”. En caso de que el Servidor Bodeguero no esté disponible (Bodeguero = 0), se debe incrementar la Cola Servidor Bodega (ColaBodega += 1), la variable “ColaBodega” servirá como contador, al principio de la simulación su valor inicial es cero, próximamente se reportará la Cola Servidor Bodega y el estado del sistema cambiará a “COLA BODEGA NO VACÍA”.
- El sistema con estado “COLA BODEGA VACÍA” recibe la señal “EntregadeProveedor”, posteriormente se genera el arribo del siguiente proveedor para lo cual se calcula el tiempo entre arribos de proveedores a través del procedimiento “TiempoArribodeProveedor”, luego se envía la señal “EntregadeProveedor” al mismo proceso, la señal tiene un retraso acorde al parámetro “TAP” que es el tiempo calculado mediante el procedimiento, de esta forma se produce el arribo del siguiente proveedor. A continuación, se debe preguntar si el Servidor Bodeguero está disponible (Bodeguero = 1), si esto es verdadero puesto que consecutivamente se enviará un servicio al servidor debemos indicar que el servidor no está disponible (Bodeguero = 0), luego se envía la señal “ServicioBodegaProveedor” al Servidor Bodeguero, la misma que contiene 6 parámetros que hacen referencia a la cantidad entregada por el proveedor de cada uno de los 6 productos, el estado del sistema se mantendrá como

“COLA BODEGA VACÍA”. En caso de que el Servidor Bodeguero no esté disponible (Bodeguero = 0), se debe incrementar la Cola Servidor Bodega (ColaBodega += 1), próximamente se reportará la Cola Servidor Bodega, luego se aumenta en una unidad el contador de la variable “ProveedorenCola” (ProveedorenCola +=1) indicando que hay un proveedor esperando el servicio de bodega, esta variable se utiliza con la finalidad de darles prioridad a los servicios solicitados por el proveedor con respecto a los solicitados por el cliente, el estado del sistema cambiará a “COLA BODEGA NO VACÍA”.

- El sistema con estado “COLA BODEGA VACÍA” recibe la señal “FinServicioBodeguero”, lo cual indica que el Servidor Bodeguero ha sido liberado, luego se debe poner la variable “Bodeguero” en uno (Bodeguero = 1) para indicar que el Servidor Bodeguero está disponible, el estado del sistema se mantendrá en “COLA BODEGA VACÍA”.

En el estado “COLA BODEGA NO VACÍA” pueden ocurrir 3 tipos de eventos representados mediante las señales: “PedidodeCliente”, “EntregadeProveedor” y “FinServicioBodeguero”. Los mismos que se detallan a continuación:

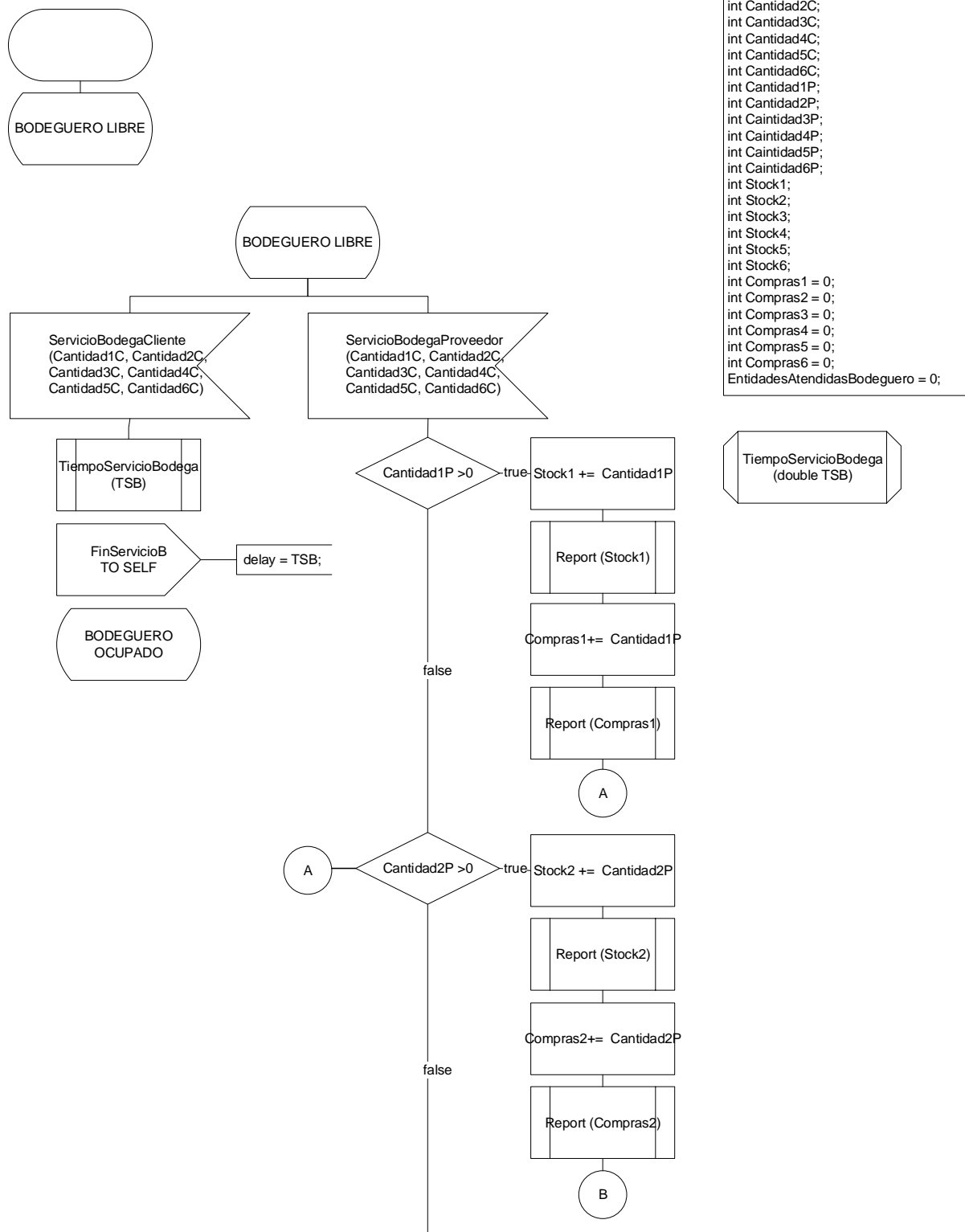
- El sistema con estado “COLA BODEGA NO VACÍA” recibe la señal “PedidodeCliente”, al estar en un estado de no vacía la llegada de una nueva señal requiriendo el servicio por parte del servidor generará que la cola aumente en una unidad (ColaBodega += 1), seguidamente se reporta la cola y el estado se mantendrá como “COLA BODEGA NO VACÍA”.
- El sistema con estado “COLA BODEGA NO VACÍA” recibe la señal “EntregadeProveedor” que produce que la cola aumente en una unidad (ColaBodega += 1), se solicita un reporte de la cola, posteriormente se aumenta en una unidad el contador de la variable “ProveedorenCola” (ProveedorenCola +=1), a continuación se calcula el parámetro TAP mediante el procedimiento “TiempoArribodeProveedor”, luego se envía la señal “EntregadeProveedor” al mismo proceso con un retraso indicado por el parámetro TAP del procedimiento, lo cual representa el arribo del siguiente proveedor, el estado del sistema se mantiene en “COLA BODEGA NO VACÍA”.

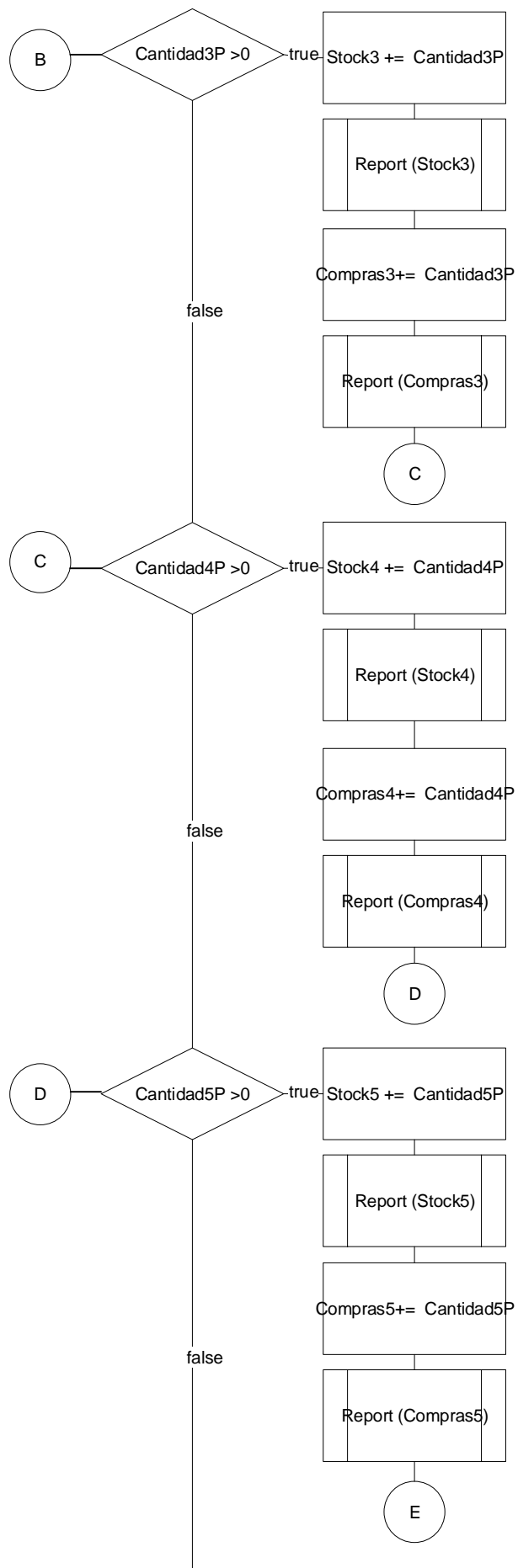


- El sistema con estado "COLA BODEGA NO VACÍA" recibe la señal "FinServicioBodeguero", se disminuye la cola en una unidad ( $\text{ColaBodega} -= 1$ ), ya que se atendería a uno de las entidades pendientes de atender que están en la cola, luego se solicita un reporte de la cola y se consulta si la cola ha quedado vacía ( $\text{ColaBodega} = 0$ ), en caso de que sea verdadero después se pregunta si la entidad que está esperando el servicio no es un proveedor ( $\text{ProveedorenCola} = 0$ ), si es verdadero se envía la señal "ServicioBodegaCliente" al "ServidorBodeguero" y el estado cambia a "COLA BODEGA VACÍA", ya que no hay ninguna entidad esperando en la cola. En caso de que la entidad esperando en la cola sea un proveedor, se envía la señal "ServicioBodegaProveedor" al "ServidorBodeguero" para que lo atienda y el estado cambia a "COLA BODEGA VACÍA". Si la cola no está vacía ( $\text{ColaBodega} \geq 1$ ) se pregunta si no hay algún proveedor en la cola esperando para ser atendido ( $\text{ProveedorenCola} = 0$ ) ya que los proveedores tienen prioridad en la atención del bodeguero, si es verdadero se envía la señal "ServicioBodegaCliente" al ServidorBodeguero y se mantiene como "COLA BODEGA NO VACÍA", si es falso se envía primero la señal "ServicioBodegaProveedor" al "ServidorBodeguero", luego se decrementa en una unidad el contador de la variable "ProveedorenCola" ( $\text{ProveedorenCola} -= 1$ ) y el estado se mantiene como "COLA BODEGA NO VACÍA".

Proceso Servidor Bodeguero

Process: ServidorBodeguero





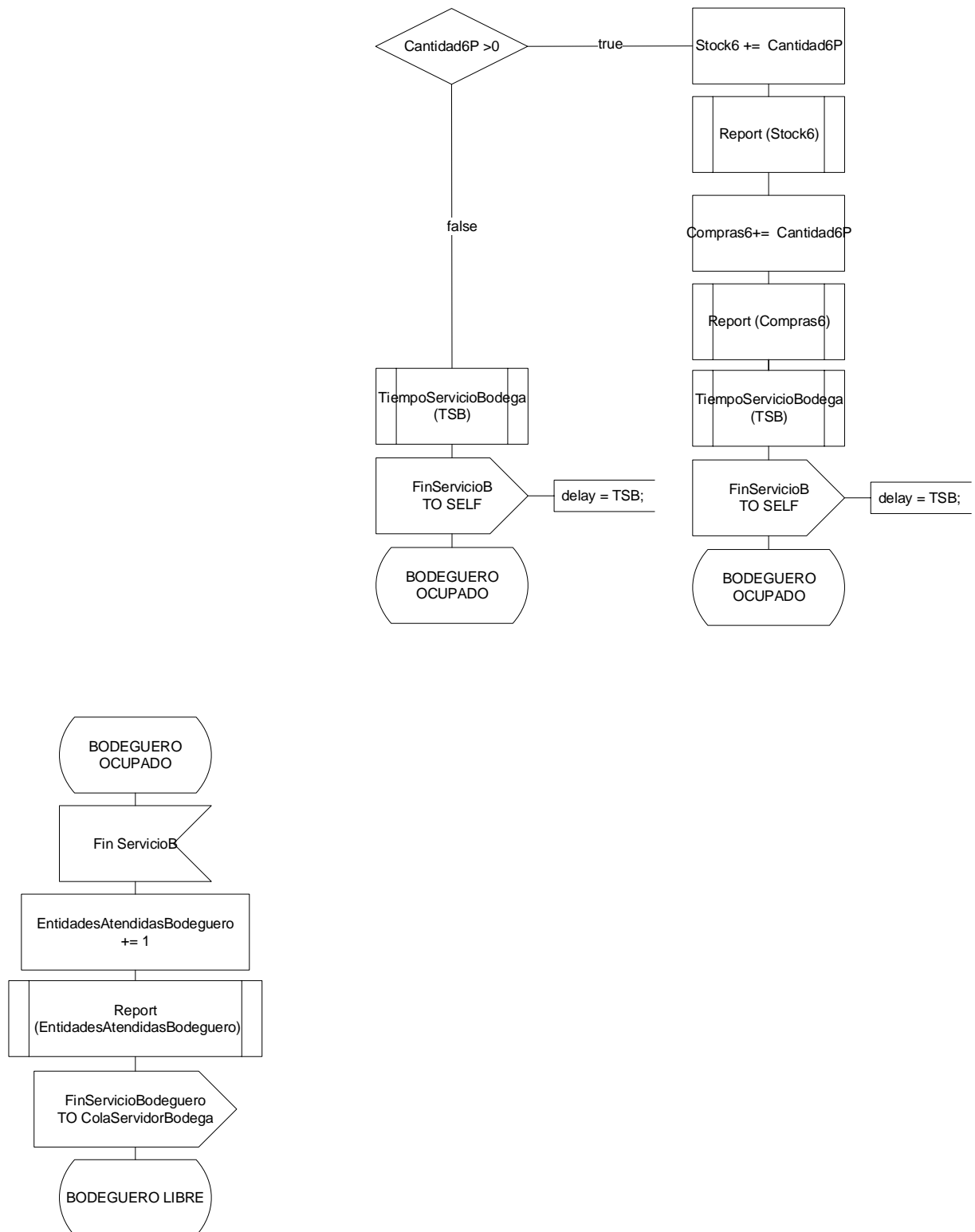


Ilustración 34. “Proceso Servidor Bodeguero”

El “ServidorBodeguero” es representado mediante los estados “BODEGUERO LIBRE” y “BODEGUERO OCUPADO”. El proceso inicia con el estado “BODEGUERO LIBRE”. En el estado “BODEGUERO LIBRE” pueden ocurrir dos tipos de eventos representados mediante

señales: “ServicioBodegaCliente” y “ServicioBodegaProveedor”. Los mismos que se detallan a continuación:

- El sistema con estado “BODEGUERO LIBRE” recibe la señal “ServicioBodegaCliente”, a continuación, se calcula el tiempo de servicio del bodeguero mediante el procedimiento “TiempoServicioBodega” que devuelve el parámetro “TSB” el mismo que es el tiempo de servicio, luego envía al mismo proceso la señal “FinServicioB” que tendrá un retraso acorde al parámetro “TSB” y durante ese tiempo se mantiene el estado como “BODEGUERO OCUPADO”.
- El sistema con estado “BODEGUERO LIBRE” recibe la señal “ServicioBodegaProveedor”, a continuación se pregunta si la cantidad del producto 1 entregada por el proveedor es mayor o igual a 1 ( $Cantidad1P > 0$ ), en caso de ser cierto se procede a aumentar en el inventario ( $Stock1 += Cantidad1P$ ) en tantas cantidades como indique el parámetro de cantidad del producto 1 “Cantidad1P”, se solicita un reporte del inventario actual del producto 1 “Stock1”, se incrementan las compras ( $Compras1 += Cantidad1P$ ) en las unidades que tiene el parámetro de cantidad del producto 1 “Cantidad1P” y se solicita un reporte de las compras; posteriormente y también en caso de ser falso que el proveedor realiza entrega del producto 1 ( $Cantidad1P = 0$ ) se pregunta si la cantidad del producto 2 entregada por el proveedor es mayor o igual a 1 ( $Cantidad2P \geq 1$ ), esta parte del proceso se repite para todos los productos seleccionados para la simulación. Luego de ser realizado para todos los productos, se determina el tiempo de servicio del bodeguero mediante el procedimiento “TiempoServicioBodega” que devuelve el parámetro “TSB”, que es el tiempo de servicio del bodeguero, luego se envía al mismo proceso la señal “FinServicioB” que tendrá un retraso acorde al parámetro “TSV” y durante ese tiempo se mantiene el estado del sistema como “BODEGUERO OCUPADO”.

En el estado “BODEGUERO OCUPADO” puede ocurrir 1 tipo de evento representado mediante la señal “FinServicioB”. A continuación, se describe el proceso:

- En el estado “BODEGUERO OCUPADO” se recibe la señal “FinServicioB”, luego se van a contar los clientes atendidos por el bodeguero con la variable “ClientesAtendidosBodeguero” que aumentará en una unidad cada vez que atienda un cliente o proveedor el bodeguero, la variable al principio de la simulación se inicializa con cero, luego se solicita un reporte del total de los clientes atendidos por el

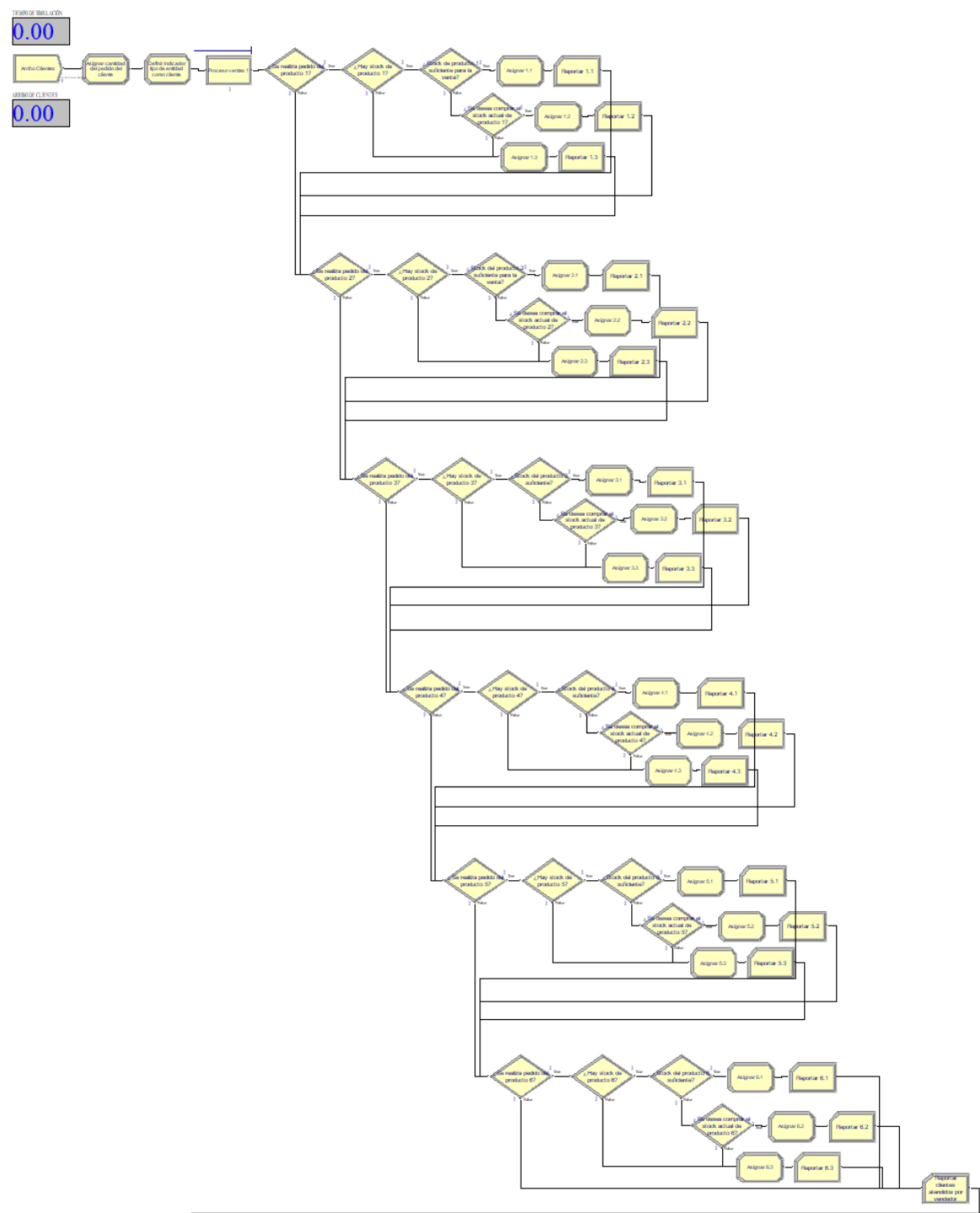
bodeguero, posteriormente se envía la señal “FinServicioBodeguero” a la Cola Servidor Bodega notificando que el servidor está desocupado y ha terminado el servicio, el estado del sistema cambia a “BODEGUERO LIBRE”.

## **Arena**

Es un software para simulación discreta basado en el lenguaje de programación SIMAN. Arena está orientado a procesos, es decir que describe como las distintas entidades se mueven a través del sistema. Los procesos se describen mediante la ubicación y conexión de módulos.

Adicionalmente es importante mencionar que es un software muy confiable para la generación de números aleatorios, se basa en un generador congruencial multiplicativo para la generación de números aleatorios uniformes entre 0 y 1.

Modelo en Arena sistema actual



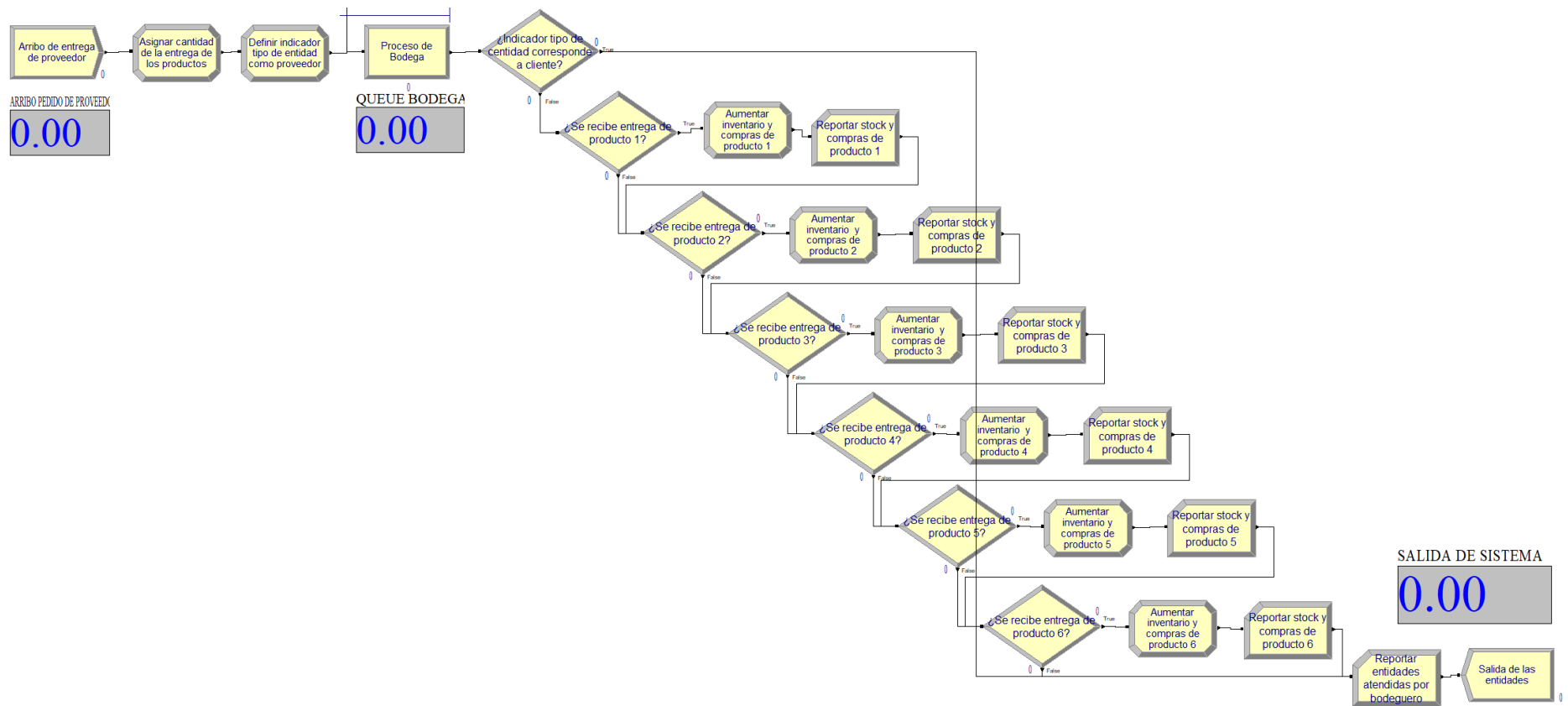


Ilustración 35, “Modelo en Arena sistema actual”



Con el módulo create generamos la llegada al sistema de los clientes al sistema, los clientes son uno de los tipos de entidades que se mueven a través del sistema. Arena generará aleatoriamente los clientes de acuerdo con la distribución tiempo entre arribo de los clientes que sigue una distribución exponencial con media de 47.80 minutos lo que se especifica también dentro del módulo create.

El cliente debe decidir la cantidad que desea comprar de cada producto esto se representa en un el módulo assign de Arena, se han creado atributos que corresponden a la cantidad, se especifica de igual forma dentro del módulo que seguirán una distribución de probabilidad discreta que ha sido creada basándose en las distribuciones empíricas de las muestras.

Con la finalidad de darles a los proveedores una prioridad en la atención del servidor que pertenece a la bodega se utiliza un nuevo módulo assign con el que vamos a asignar al cliente un atributo al cual se ha denominado indicador tipo de entidad que para los clientes tendrá un valor de 2, más adelante también se definirá también el indicador tipo de entidad de los proveedores.

Los clientes serán atendidos primero por el servidor vendedor representado mediante el módulo process de Arena, este módulo también representa la cola del servidor vendedor y la interacción entre la cola y el servidor, es decir si el servidor no está ocupado puede recibir un cliente y brindarle el servicio, si por el contrario el servidor está ocupado ya que se encuentra brindándole el servicio a un cliente, no puede recibir otro cliente hasta que termine de dar el servicio y sea liberado. Si cuando arriba un nuevo cliente, el vendedor está disponible ese cliente pasará a ser atendido por el servidor, si el vendedor no está disponible, se aumenta la cola en una unidad. El sistema estará en estado "COLA VENDEDOR VACÍA" cuando haya cero clientes esperando en la cola y en estado "COLA VENDEDOR NO VACÍA" cuando haya uno o más clientes esperando en la cola, así mismo el sistema estará en estado "VENDEDOR LIBRE" cuando el vendedor no esté atendiendo ningún cliente y en estado "VENDEDOR OCUPADO" cuando el vendedor este atendiendo un cliente. Los clientes esperando en la cola serán atendidos según el método FIFO (first in first out) y no es necesario solicitar un reporte de la cola como en SDL porque al final de la simulación Arena muestra indicadores del tiempo medio de los clientes esperando en la cola y el promedio de clientes en la cola. El tiempo que se demora el vendedor en atender un cliente sigue una exponencial con media igual a 10 minutos, esto de igual forma es especificado dentro del módulo process con la finalidad de que se generen números aleatorios que definan el tiempo de servicio para cada cliente atendido por el servidor.

A modo de ejemplo para la siguiente explicación se tomará como referencia el producto 1, con el módulo decide de Arena que implica una decisión doble vía basada en una condición se consulta si el cliente desea realizar un pedido del producto 1, es decir si el atributo cantidad del producto 1 es mayor a cero, si es verdadero luego se pregunta mediante otro módulo decide con la condición de si existe inventario del producto 1 lo que implicaría que el inventario del producto 1 sea mayor a cero, si es cierto se procede a consultar con un módulo decide basado en la condición de si existe inventario suficiente del producto 1 para realizar la venta esto es que el inventario del producto 1 sea mayor o igual que el atributo de cantidad del producto 1, en caso de ser verdadero ya que se concreta la venta se realizan las siguientes asignaciones mediante el módulo assign (Asignar 1.1): se incrementan las ventas del producto 1 conforme a las unidades vendidas (atributo de la cantidad del producto 1) y se disminuye el inventario del producto 1 en las unidades vendidas (atributo de la cantidad del producto 1), posteriormente con la ayuda del módulo record (Reportar 1.1) se solicitan los siguientes reportes: el reporte de ventas del producto 1, el reporte del inventario del producto 1 y el reporte del contador de clientes satisfechos del producto 1, este último previamente al reporte aumenta en una unidad debido a que se vende en su totalidad las unidades que el cliente desea comprar, esto lo realiza automáticamente el módulo report. Si la decisión del módulo decide que pregunta si existe inventario suficiente del producto 1 para realizar la venta es falsa se procede a consultar con un módulo decide basado en una decisión por casualidad si el cliente desea comprar lo que se dispone actualmente en el inventario del producto 1, en este módulo el 98% de las veces el resultado es verdadero y el 2% de las veces el resultado es falso, en caso de ser verdadero se realizan las siguientes asignaciones a través del módulo assign (asignar 1.2): se incrementan las ventas del producto 1 conforme a las unidades vendidas (variable inventario del producto 1), se aumentan las ventas perdidas del producto 1 (el atributo cantidad del producto 1 menos la variable inventario del producto 1), se disminuye el inventario del producto 1 a cero, luego a través del módulo record (Reportar 1.2) se solicitan los reportes que se detallan a continuación: el reporte de ventas del producto 1, el reporte del inventario del producto 1, el reporte de las ventas perdidas del producto 1 y el reporte del contador de clientes satisfechos del producto 1, el contador de clientes satisfechos del producto 1 antes de reportarse aumenta en una unidad a causa de que existen ventas perdidas esto lo realiza automáticamente el módulo report. Si la decisión de consultar si el cliente desea comprar lo que se dispone actualmente en el inventario del producto 1 es falsa o si la decisión de preguntar si existe inventario del producto 1 es falsa se procede con el módulo assign (Asignar 1.3) a incrementar las ventas perdidas en las unidades que no se pudieron vender por falta de stock (atributo de la cantidad del producto 1), posteriormente se solicitan mediante el módulo report los reportes detallados: el reporte de las ventas perdidas

del producto 1 y el reporte del contador de clientes satisfechos del producto 1, el último antes de reportarse aumenta en una unidad a causa de que hubieron ventas perdidas.

Luego de los módulos record: Reportar 1.1, Reportar 1.2, Reportar 1.3 y también en el caso de que el cliente no solicite comprar el producto 1, es decir si cuando se consulta si el cliente desea realizar un pedido del producto 1, la respuesta es falsa entonces se consulta si el cliente desea realizar un pedido del producto 2, se repite entonces el mismo proceso del producto 1 con el producto 2 y posteriormente con los demás productos que se incluyen en la simulación. Posteriormente mediante el módulo record se cuenta y se reportan los clientes atendidos por el vendedor. Todo lo descrito hasta aquí en SDL equivaldría a los procesos: “ColaServidorVendedor” y “ServidorVendedor”.

El arribo de las entregas de los proveedores se representa con el módulo create que genera aleatoriamente la llegada de los pedidos al sistema, que son el otro tipo de entidad que se mueve a través del sistema. Arena generará aleatoriamente el arribo de los pedidos de los proveedores de acuerdo con la Distribución Exponencial con media igual 32 horas lo que también se especifica dentro el módulo create.

Se especifica la cantidad de cada producto que llega en la entrega del proveedor a través del módulo assign de Arena, donde se define la cantidad como un atributo. Se define también dentro del módulo assign que las cantidades de los productos siguen una distribución de probabilidad discreta que ha sido creada basándose en las distribuciones empíricas de las muestras.

Con la finalidad de darles a los proveedores una prioridad en la atención del servidor de bodega se utiliza un nuevo módulo assign con el que se asigna al pedido del proveedor el atributo indicador tipo de entidad con el valor de 1.

Las entidades, clientes y proveedores son atendidos por el servidor bodeguero, mediante el módulo process de Arena se representa al bodeguero, este módulo también representa la cola del servidor vendedor y la interacción entre la cola y el servidor, es decir si el servidor no está ocupado puede recibir un cliente y brindarle el servicio, si por el contrario el servidor está ocupado ya que se encuentra brindándole el servicio a un cliente, no puede recibir otro cliente hasta que termine de dar el servicio y sea liberado. Si cuando arriba un nuevo cliente el vendedor está disponible ese cliente pasará a ser atendido por el servidor, si el vendedor no está disponible, se aumenta la cola en una unidad. El sistema estará en estado “COLA BODEGA VACÍA” cuando haya cero entidades esperando ser atendidas en la cola y en estado

“COLA BODEGA NO VACÍA” cuando haya uno o más clientes esperando en la cola. Así mismo el sistema estará en estado “VENDEDOR BODEGUERO LIBRE” cuando el bodeguero no esté atendiendo una entidad y “BODEGUERO OCUPADO” cuando el bodeguero esté atendiendo una entidad. Adicionalmente se ha definido en la cola la prioridad de atención a las entidades con menor valor del atributo indicador tipo de entidad, que serían los proveedores ya que se definió previamente su indicador con el número 1, mientras que los clientes tienen el número 2. En Arena no es necesario solicitar un reporte de la cola como en SDL porque al final de la simulación Arena muestra indicadores del tiempo medio de las entidades esperando en la cola y el promedio de las entidades en la cola. El tiempo que se demora el vendedor en atender un cliente sigue una exponencial con media igual a 12 minutos, esto de igual forma es especificado dentro del módulo process con la finalidad de que se generen números aleatorios que definan el tiempo de servicio para cada cliente atendido por el servidor.

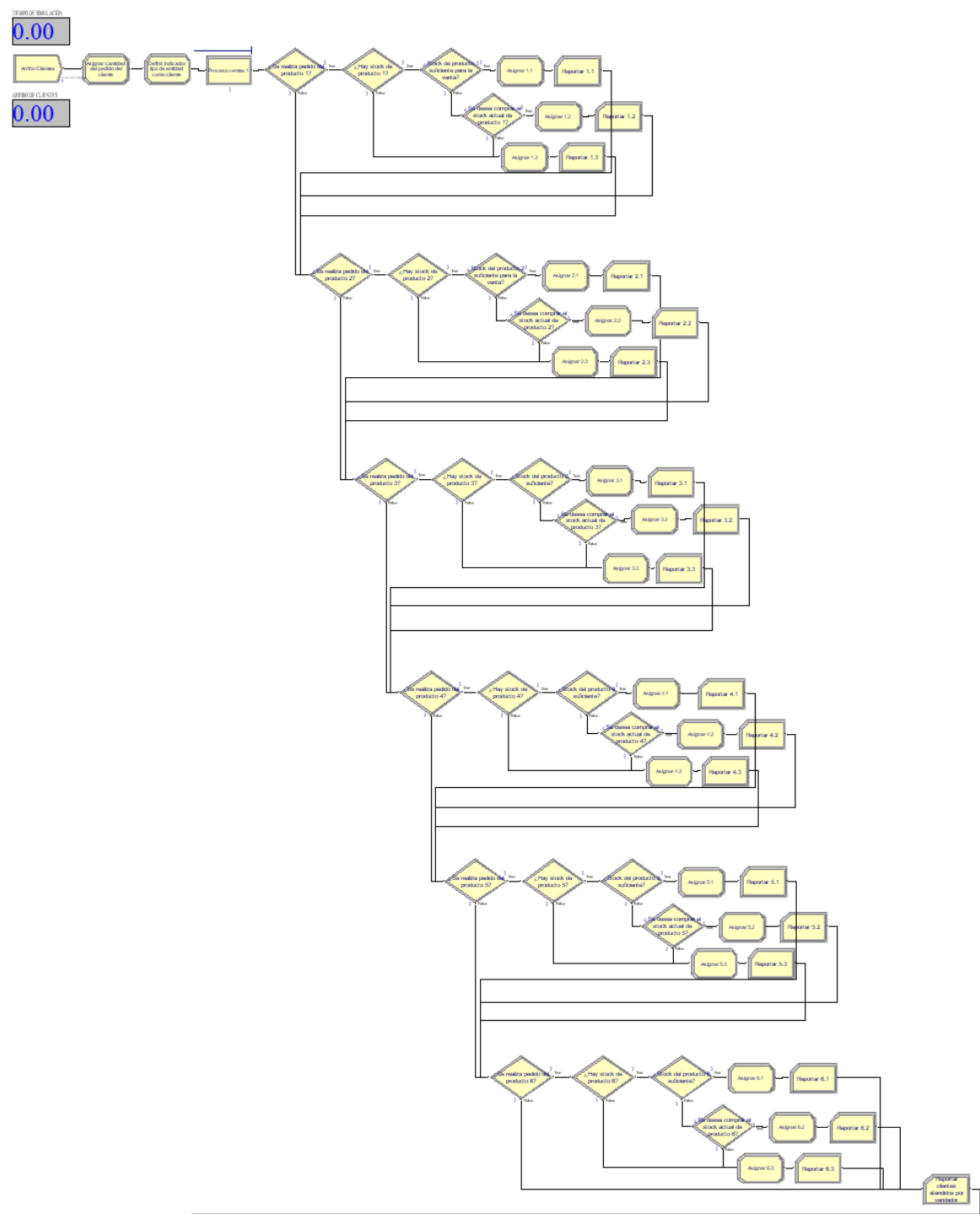
Con el módulo decide se pregunta si el atributo indicador tipo de entidad corresponde a clientes en caso de ser cierto es decir en caso de que sea 2, luego se reporta que el bodeguero ha atendido al cliente por medio del módulo report y posteriormente el cliente sale del sistema con el módulo dispose, Si la pregunta de si el atributo indicador tipo de entidad corresponde a clientes es falsa, mediante un módulo decide se pregunta si se recibe una entrega del producto 1, es decir si el atributo de cantidad del producto 1 es mayor a cero, en caso de ser verdadero se procede mediante el módulo assign a aumentar la cantidad del inventario del producto 1 en la cantidad de unidades compradas (atributo de cantidad del producto 1) y a incrementar las compras en la cantidad de unidades compradas (atributo de cantidad del producto 1), posteriormente se solicita un reporte con el módulo record del stock del producto 1 y de las compras del producto 1. En caso de que la pregunta de si se recibe una entrega del producto 1 sea falsa se procede mediante un módulo decide a preguntar si se recibe una entrega del producto 2, para el producto 2 y los demás productos incluidos en la simulación se realiza el mismo proceso que con el producto 1. Luego se reporta mediante un módulo record que la entidad ha sido atendida por el bodeguero y finalmente la entidad abandona el sistema por medio del módulo dispose de Arena.

En la configuración de la corrida especificamos que se simula un año laboral compuesto de 280 días (2240 horas) con  $n=101$  réplicas piloto. Se procedió a calcular a partir de los datos del sistema actual un intervalo para la media del nivel de servicio con un nivel de confianza del 95%, el intervalo es (68.99%, 72%), el intervalo está a menos del 5% de la media muestral por lo que se considera que no se necesitan más replicas adicionales.

## Modelo de sistema de control de inventario periódico

El sistema de control de inventario periódico se basa en realizar un control del inventario en intervalos de periodos fijos (P), para realizar un nuevo pedido se evaluará la posición actual del inventario (I), que es el inventario actual más los pedidos en tránsito en caso de existir y la cantidad a ordenar (Q) será el nivel de inventario objetivo (T) menos la posición actual del inventario  $Q = T - I$ , es decir que la cantidad a ordenar en cada periodo será variable. Se utiliza la siguiente fórmula para calcular el nivel de inventario objetivo,  $T = D * (T + L) + Z * S * \sqrt{T + L}$ , donde D es la demanda promedio diaria, P es el intervalo de revisión (periodo en días), L es el tiempo de entrega del pedido (en días), Z es el factor de seguridad que se calcula tomando en consideración el nivel de servicio deseado y S es la desviación estándar de la demanda diaria. Nótese que debido a que la empresa tiene una demanda variable dentro del nivel de inventario objetivo se ha considerado el cálculo del inventario de seguridad,  $M = Z * S * \sqrt{T + L}$ . El nivel de servicio se define como  $NS = \text{clientes satisfechos} \div \text{clientes totales}$  y es importante acotar que mientras mayor sea el nivel de servicio que se desea alcanzar mayor será el inventario de seguridad que la empresa debe disponer.

Modelo en Arena sistema mejorado



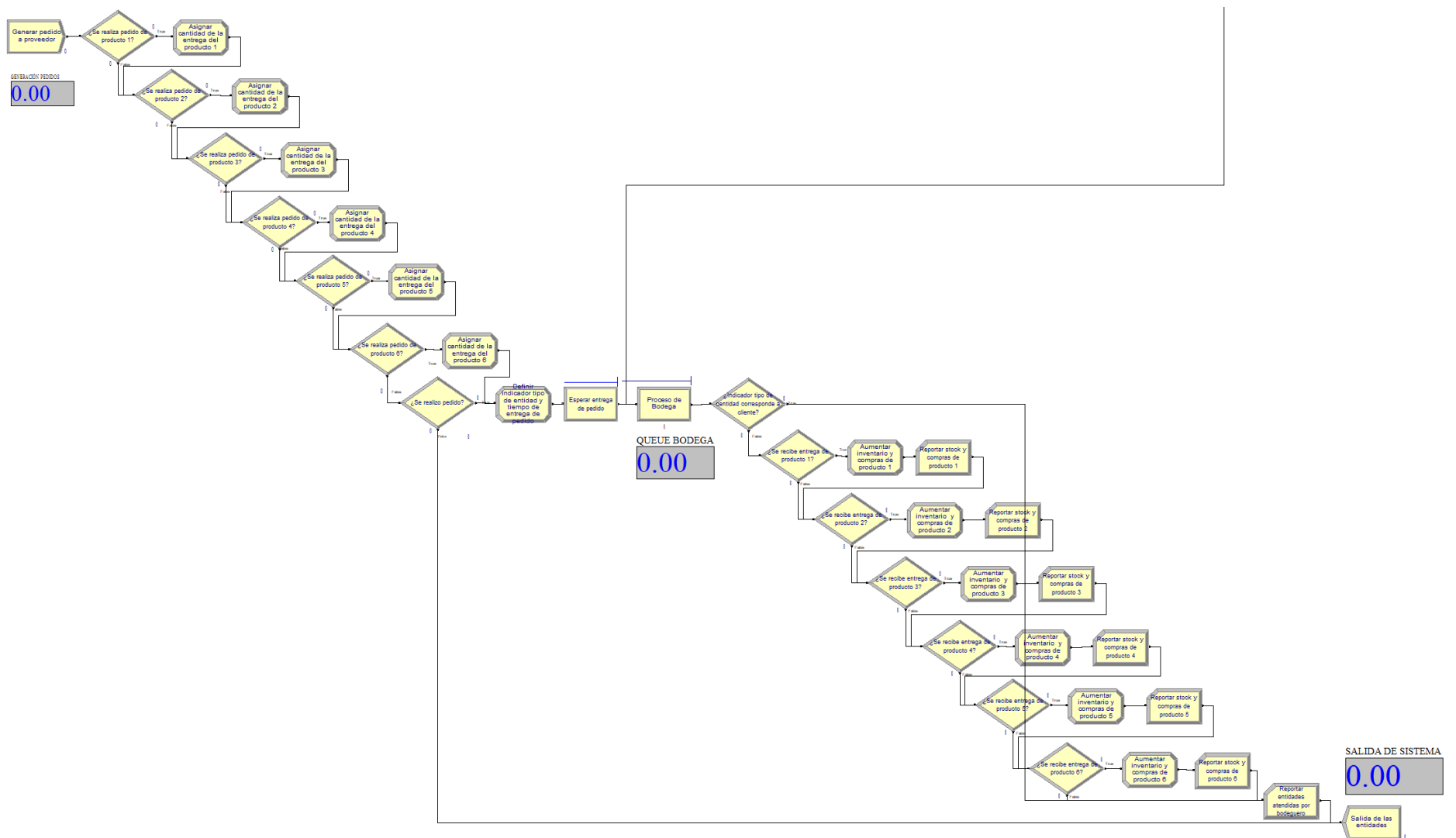


Ilustración 36, “Modelo en Arena sistema mejorado”

En el nuevo sistema propuesto se requiere un nivel de inventario objetivo (T) para cada producto a simular adicionalmente es importante denotar que los pedidos se generarán en periodos fijos (P), producto de estas nuevas condiciones se han implementado ciertos cambios al modelo original.

Se describirá básicamente la parte del modelo que se modificó, es decir la parte del arribo de las entregas de los proveedores, que se representa con el módulo create que genera los pedidos en el sistema de forma constante, según la cantidad de días del periodo, con el módulo decide que implica una decisión doble vía basada en una condición se consulta si se realiza pedido del producto 1, esta pregunta está basada en la condición de si la variable del inventario del producto 1 es menor que el nivel de inventario objetivo del producto 1, en caso de ser verdad luego se especifica la cantidad de producto 1 que se pide mediante el módulo assign, la cantidad pedida va a ser igual al nivel de inventario objetivo del producto 1 menos la variable inventario del producto 1, en caso de realizar o no realizar pedido del producto 1, se procede a consultar si se realiza pedido del producto 2, luego de esto se realizará con todos los productos simulados el mismo proceso que con el producto 1, si no se realiza pedido de ningún producto la entidad saldrá del sistema mediante el módulo dispose. En caso de que se realice un pedido, se procede con un assign a definir el atributo indicador tipo entidad que es 1 por ser proveedor y también el atributo tiempo de entrega, en el caso de COFEGO el tiempo entrega es 2 días por lo que se define el atributo como  $TNOW (\text{tiempo de simulación}) + 16$ , ya que son 16 horas laborables en los 2 días, luego el pedido entrará al módulo hold que lo retendrá hasta que se cumpla la condición, en la condición se especifica que el  $TNOW$  sea igual al atributo tiempo de entrega. Con estas modificaciones realizadas al modelo original se logra incorporar los cambios originados por la implantación del sistema de inventario de revisión periódica.

En la configuración de la corrida especificamos que se simula un año laboral compuesto de 280 días (2240 horas) con  $n = 10$  réplicas piloto.

## **Diseño de experimentos**

Ya que la cantidad de productos a comprar (Q) toma distintos valores que dependen del intervalo del periodo (P) y del nivel de servicio deseado, se definen estos como factores para el diseño de experimentos. Los niveles fijados para el periodo son 5 y 10 días, para determinarlos se tuvo en cuenta la opinión del Gerente General quien se encarga de la gestión



de compras en la actualidad. Los niveles de servicio deseados se definieron en 90% y 97.5%. Se procede a realizar un diseño factorial, el nivel “–” del factor periodo es 5 y el nivel “+” del factor periodo es 10, el nivel “–” del factor nivel de servicio deseado es 90% y el nivel “+” del factor periodo es 97.5%, la variable que mide el sistema es el nivel de servicio real esta sería mi variable dependiente, se realizan 10 réplicas de cada grupo. Se muestra a continuación el diseño del experimento.

Nivel servicio deseado	Periodo
90%	5
90%	10
97.50%	5
97.50%	10

Ilustración 37, “Diseño de experimentos”

A continuación, se detallan los resultados de cada una de las réplicas y la media de cada grupo.

Réplica										Nivel de servicio medio
1	2	3	4	5	6	7	8	9	10	
97.49%	97.66%	98.39%	98.06%	96.11%	97.05%	97.55%	97.19%	98.55%	96.55%	97.46%
96.37%	97.15%	96.98%	96.16%	96.50%	97.92%	97.51%	95.29%	97.51%	95.08%	96.65%
98.81%	98.35%	99.73%	98.81%	99.10%	99.36%	98.73%	98.55%	99.64%	98.59%	98.97%
98.59%	99.52%	99.69%	99.28%	98.91%	99.75%	99.20%	99.60%	99.50%	99.72%	99.38%

Ilustración 38, “Replicas del diseño de experimentos”

Con las 10 réplicas piloto para cada grupo se elaboró un intervalo para la media del nivel de servicio con un nivel de confianza del 95%, los intervalos de confianza se detallan a continuación:

		Intervalo de confianza	
Nivel servicio deseado	Periodo	Intervalo inferior	Intervalo superior
90%	5	96.91%	98.01%
90%	10	95.97%	97.33%
97.50%	5	98.63%	99.30%
97.50%	10	99.10%	99.65%

Ilustración 39, “Intervalos de confianza”

Puesto que todos los intervalos están a menos del 5% de la media muestral se considera que no se necesitan replicas adicionales.

## ANOVA 2 factores fijos

### Análisis de la tabla ANOVA

Con la finalidad de determinar si alguno de los factores (periodo o nivel de servicio deseado) o la interacción entre ambos factores influye en la variable dependiente (nivel de servicio), se procede a realizar la tabla ANOVA. Notación  $Y_{ijn} = \alpha_i + \beta_j + \alpha\beta_{ij} + e_{ijn}$ , donde  $\alpha_i$  es el efecto del i-ésimo nivel del factor periodo agrupando los niveles del factor nivel de servicio deseado con  $i=1,2$ ;  $\beta_j$  es el efecto del j-ésimo nivel del factor nivel de servicio deseado con  $j=1,2$  agrupando los niveles del factor periodo;  $\alpha\beta_{ij}$  es el efecto de la interacción del i-ésimo nivel del factor periodo y del j-ésimo nivel del factor nivel de servicio deseado,  $e_{ijn}$  es el error aleatorio de la n-ésima replica de la combinación del i-ésimo nivel del factor periodo y del j-ésimo nivel del factor nivel de servicio deseado factor donde  $n = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10$ . A continuación se presenta la tabla ANOVA.

Fuente de variación	Grados de libertad	Suma de cuadrados	Media de cuadrados	Estadístico F	P valor
Periodo	1	0.0000403	0.0000403	0.8656	0.358364
Nivel servicio esperado	1	0.0044898	0.0044898	96.3266	1.02E-11
Nivel servicio esperado * Periodo	1	0.0003739	0.0003739	8.0211	0.007525
Residuo	36	0.0016780	0.0000466		

Ilustración 40, "Tabla ANOVA"

Se observa que la interacción de ambos factores es significativa, ya que su p valor es menor a 0.05, por lo que se rechaza  $H_0: \alpha\beta_{11} = \alpha\beta_{12} = \alpha\beta_{21} = \alpha\beta_{22} = 0$  se procede a realizar el test de Tukey para determinar las combinaciones de los niveles de los factores en las que existen diferencias, el resumen que muestra R es el siguiente:

Grupos	Intervalo			
Periodo : Nivel servicio esperado	Diferencia	Inferior	Superior	P valor
10:90 - 5:90	-0.00812311	-0.016346078	9.99E-05	0.0538744
5:97.5 - 5:90	0.01507471	0.006851742	2.33E-02	0.0001036
10:97.5 - 5:90	0.01918048	0.010957516	2.74E-02	0.0000017
5:97.5 - 10:90	0.02319782	0.014974853	3.14E-02	0
10:97.5 - 10:90	0.02730359	0.019080627	3.55E-02	0
10:97.5 - 5:97.5	0.00410577	-0.004117193	1.23E-02	0.5413096

Ilustración 41, "Test de Tukey"

Se observa que las interacciones 5:97.5 - 5:90, 10:97.5 - 5:90, 5:97.5 - 10:90, y 10:97.5 - 10:90 son significativas por lo que existen diferencias entre los niveles comparados y se puede

afirmar que la interacción 10:97.5 es mejor que 5:90 y que 10:90, que la interacción 5:97.5 es mejor que 5:90 y 10:90, mientras que entre las interacciones 10:97.5 - 5:97.5 y 5:90 - 10:90 no hay diferencias entre los niveles comparados.

## Validación de hipótesis del ANOVA

Las réplicas de los grupos fueron aleatoriamente generadas por el generador de números aleatorios del software Arena, lo que indica que se cumple la hipótesis de independencia. Se puede observar que la gráfica de residuos versus valores predichos muestra problemas con homocedasticidad, adicionalmente se realizó el test de Levene el cual arroja un p valor de 0.04605 que al ser menor que 0.05 rechaza  $H_0$ : Las varianzas entre los grupos son iguales, se procedió a aplicar una transformación logarítmica a la variable dependiente sin embargo el problema persiste. Considerando que en este diseño de experimentos se realizaron 10 réplicas para cada grupo con lo cual se tiene un diseño balanceado y ya que se conoce que los diseños balanceados ayudan a mitigar los problemas de heteroscedasticidad, sumado al hecho de que en la gráfica no se observa una muy exorbitante violación de la homocedasticidad y de que el p valor de test de Levene es muy cercano a 0.05 se da por validada la hipótesis.

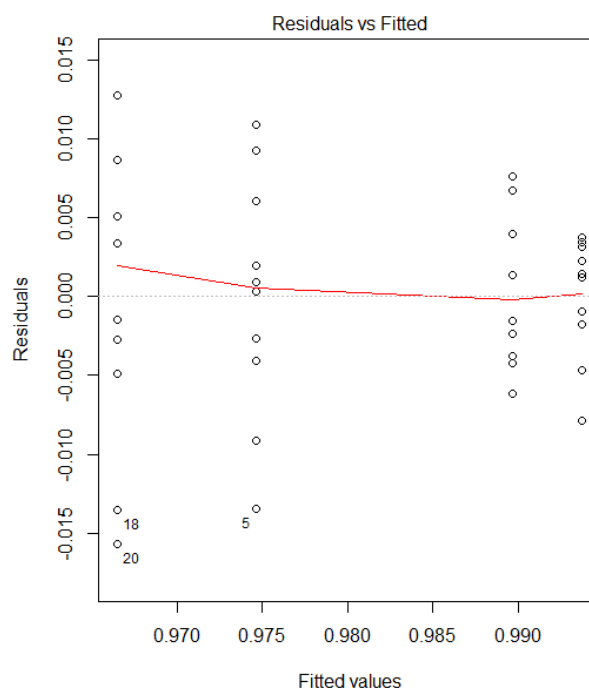


Ilustración 42, “Gráfico residuos versus valores predichos”

Se observa en la gráfica que los residuos se distribuyen como una normal, adicionalmente se aplicó el Test de Shapiro – Wilk, el p valor es de 0.6263, ya que es mayor que 0.05 no rechaza  $H_0$ : Los datos provienen de una distribución normal, por lo tanto, se da por validada la hipótesis de normalidad.

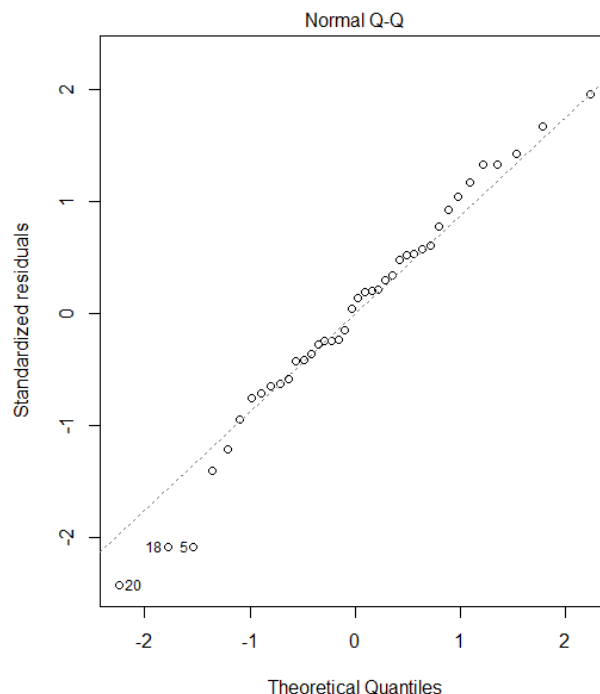


Ilustración 43, “QQ Plot”

## Verificación y validación

En cuanto a validación conceptual del modelo, se procedió a revisar el modelo con el Ing. Fernando Gonzales, Gerente General quien es la persona que más conocimiento del negocio tiene, se procedió a explicarle el modelo y se le consulto si le parecía razonable tanto lógicamente como conceptualmente, a lo cual asintió. Adicionalmente también se validó con el modelo de simulación de forma gráfica, mediante la animación de Arena, se procedió a mostrar gráficamente como es el movimiento de las entidades (clientes y proveedores) a través del sistema, de igual forma se mostraron gráficos para los recursos (vendedores y bodegueros) en los distintos estados (ocupado y libre) que experimentan mientras brindan su servicio a las entidades, con la finalidad de que se pueda apreciar gráficamente la dinámica del sistema. En lo que respecta a la validación de los datos, se puede afirmar su validez puesto que la información fue proporcionada por el sistema de la empresa.

Se realizaron pruebas degeneradas, en las que se fijaba para un solo producto del sistema el atributo cantidad de las ventas y el atributo cantidad de las compras en 200 y 1000

respectivamente, mientras que los atributos de la cantidad de compras y ventas de los demás productos se fijaron a cero, con esto se analizaba mediante los reportes que imprime Arena en Excel como iban variando las distintas variables del sistema: clientes satisfecho, clientes no satisfechos, compras, ventas, inventario, etc; las pruebas se realizaron para 10 días y se aplicó lo mismo para los 6 productos simulados en el modelo del sistema real, todos los resultados fueron buenos y precisos.

Se realizaron pruebas de condición extrema para el modelo mejorado fijando el inventario inicial de los productos en un valor extremadamente alto 1,000,0000; con la finalidad de verificar que el sistema no genere compras de estos productos ya que su inventario está por encima del nivel del inventario objetivo, se simularon 280 días todos los resultados fueron buenos y precisos, al verificar el reporte de compras se constató que las compras fueran 0.

Al simular el sistema actual se observa que las salidas del sistema son coherentes con la realidad de la empresa, se evidencian ventas perdidas y un bajo nivel de servicio que son las principales preocupaciones de la empresa y el motivo por el que es necesario la implementación de un sistema de inventario, con lo cual se verifica que se cumple el objetivo de predecir el comportamiento del sistema.

La verificación se constató con la correcta implementación del modelo conceptual en el software de simulación, adicionalmente es importante recalcar que Arena dispone de un generador de números pseudo aleatorios que es muy bueno y confiable, lo cual también es muy importante para dar por verificado el modelo.

## **Recomendaciones y conclusiones**

- El modelo conceptual constituye una especificación del modelo, la cual será útil independientemente de la herramienta utilizada para la simulación.
- El modelo conceptual realizado en lenguaje SDL especifica detalladamente mediante un lenguaje no ambiguo las distintas relaciones dinámicas que existen en el sistema, lo cual permite que todos los interesados en el modelo lo comprendan de forma sencilla y clara.

- El modelo conceptual proporcionará a la empresa la facilidad de adaptar el modelo de simulación a distintos escenarios que le interese evaluar.
- El ANOVA revela que las mejores combinaciones de los factores son: un periodo de 10 días con un nivel de servicio esperado de 97.5% y un periodo de 5 días con un nivel de servicio esperado de 97.5%.
- El nivel de servicio por la implementación del sistema de revisión de inventario periódico mejoró, el intervalo de confianza para la media del nivel de servicio antes de la implementación del sistema de inventario era (68.99%, 72%), posterior a la implementación se observa que es (99.10%, 99.65%) con un nivel de servicio deseado del 97,5% y un periodo de 10 días y (98.63%, 99.30%) con un nivel de servicio deseado del 97,5% y un periodo de 5 días.
- Considerando la variabilidad de la demanda del sistema se considera que los intervalos de confianza dan indicios de un buen nivel de servicio.
- Los resultados han sido orientados al nivel de satisfacción del cliente.
- La simulación está orientada a simular un año de operaciones por lo tanto los mejores escenarios se han definido considerando ese periodo de tiempo y los datos de entrada para el modelo fueron los correspondientes al año 2017.
- Se recomienda recoger datos como el tiempo entre de arribo de clientes, tiempo de servicio de vendedor, tiempo de servicio de bodeguero, probabilidad de compra de clientes cuando no hay inventario suficiente para atender su pedido, etc.; de modo general los datos que se han supuesto y no han sido validados para obtener un modelo más realista que permita obtener resultados más precisos.
- Se recomienda extender el análisis a los demás productos que oferta la empresa, como mínimo debería implementarse para los 30 productos que corresponden al 80.38% de las ventas anuales.
- Se recomienda realizar el análisis factorial con más niveles de los factores a fin de buscar otras alternativas que podrían ser factibles.

## **Bibliografía**

Fonseca, Pau; 2014, Formal Languages for Computer Simulation, pp. 145 – 204

Burak, Buke; 2016, Simulation, pp. 27 -37

Sargent, RG; 2013, Verification and Validation of simulation models

DeGroot, Morris; 2012, Probability and Statistics

Guerrero, Humberto; segunda edición, Inventarios Manejo y Control

Lalane, Christophe; 2006, R Companion to Montgomery's Design and Analysis of Experiments